



# Green-Expert - Un solveur généralisé associé à un générateur de formulations pour la méthode des intégrales de frontières

Patrick Sztulzaft

## ► To cite this version:

Patrick Sztulzaft. Green-Expert - Un solveur généralisé associé à un générateur de formulations pour la méthode des intégrales de frontières. Electromagnétisme. INSTITUT NATIONAL POLYTECHNIQUE DE GRENOBLE, 1994. Français. NNT: . tel-01331763

**HAL Id: tel-01331763**

**<https://hal.science/tel-01331763>**

Submitted on 14 Jun 2016

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# THÈSE

présentée par

**Patrick SZTULZAF**

Pour obtenir le diplôme de Docteur de  
L'INSTITUT NATIONAL POLYTECHNIQUE DE GRENOBLE  
(Arrêté ministériel du 30 mars 1992)  
Spécialité : Énergétique Physique

---

## GREEN-EXPERT

### UN SOLVEUR GÉNÉRALISÉ ASSOCIÉ À UN GÉNÉRATEUR DE FORMULATIONS POUR LA MÉTHODE DES INTÉGRALES DE FRONTIÈRES

---

Date de soutenance : le 11 février 1994

#### *Composition du jury*

Président	Monsieur	E. PIC
Rapporteurs	Messieurs	A. RAZEK L. KRÄHENBÜHL
Examineurs	Messieurs	J.M. DEDULLE P. MASSÉ

Thèse préparée au sein du laboratoire Madylam

## MERCI À ...

Philippe MASSÉ, qui m'a proposé ce sujet, et qui a toujours su guider sans imposer, expliquer sans s'énerver, écouter et aider. Merci aussi pour cette disponibilité constante qui m'a bien souvent permis de sortir des moments difficiles de ces quatre années de travail. Merci enfin pour m'avoir aidé à comprendre qu'une des vertus de la recherche était l'humilité.

Monsieur Étienne PIC, directeur du Laboratoire d'Électromagnétisme et de Micro-Ondes (LEMO), qui a bien voulu présider le jury ainsi que messieurs Pierre SAGUET et André ROGER qui m'ont donné dans ce laboratoire le goût de la modélisation numérique en encadrant respectivement mon stage ingénieur et mon DEA.

Messieurs Laurent KRÄHENBÜHL et Adel RAZEK qui ont accepté de faire partie du jury et d'être les rapporteurs de ce mémoire. Merci aussi à Laurent KRÄHENBÜHL pour les discussions fructueuses que nous avons eues.

Thierry BREVILLE, PDG de DT2i, qui m'a fait confiance et a continué à financer ce projet à l'issu du contrat initial.<sup>1</sup>

MADYLAM qui a bien accepté un chercheur dont le bureau fut vide la plupart du temps, ce qui n'a pas empêché l'excellent contact que j'ai eu avec toutes les personnes que j'ai pu y rencontrer.

Tous les salariés de DT2i, anciens et présents, que j'ai côtoyé et qui ont contribué à l'ambiance de travail dans laquelle j'ai évolué. Merci particulièrement aux membres de l'équipe "Simul" : Alain GLIÈRE et Laurent CAILLOT qui ont guidé mes premiers pas dans l'utilisation de Flux-Expert et le développement informatique; Philippe ALEXANDRE avec qui j'ai souvent eu des discussions instructives; Pascale PHAM qui m'a souvent expliqué la physique des phénomènes que je cherchais à modéliser; Jean-Marc DEDULLE qui a accepté de représenter DT2i au jury et qui lui aussi n'a pas perdu patience en m'expliquant les subtilités de l'induction; et enfin merci à Jean-François MISSONIER et Frédéric KUNEGEL qui m'ont permis de travailler sans subir la plupart des contraintes industrielles qu'ils assument et avec qui j'ai eu des discussions éclairantes sur les subtilités du Fortran. Enfin, à tous, merci pour leur bonne humeur.

Tous ceux qui m'ont supporté et aidé durant ces années de recherche et plus particulièrement Delphine, ma compagne.

---

<sup>1</sup> Ce travail fut financé pendant trois ans par un contrat CIFRE entre l'entreprise DT2i (38240 MEYLAN) et le laboratoire MADYLAM (38402 ST MARTIN D'HÈRES). Il fut prolongé un an à la charge seule de DT2i.

<b>INTRODUCTION</b>	<b>4</b>
<b>CHAPITRE 1 : PRÉSENTATION DE LA MIF</b>	<b>6</b>
1 Introduction .....	6
2 Aspects fondamentaux .....	7
2.1 Préliminaires .....	7
2.2 Position du problème .....	7
2.3 Formulation Faible .....	8
2.4 La méthode des résidus pondérés .....	10
2.5 Solution fondamentale .....	11
2.6 Fonction de Green .....	13
3 L'exemple complet du Laplacien .....	15
4 Champs d'application de la MIF .....	18
5 Aspects complémentaires .....	20
5.1 Hétérogénéité du milieu traité .....	20
5.2 Termes sources .....	21
5.3 Non-linéarités .....	23
5.4 Les singularités .....	24
<b>CHAPITRE 2 : PRINCIPES DE BASE</b>	<b>29</b>
1 Introduction .....	29
2 La simulation numérique .....	29
3 Les diverses approches .....	31
4 Principes du couple .....	34
5 Décomposition analytique d'une résolution d'un problème intégrales de frontières	38
5.1 Préliminaires .....	38
5.2 Discretisations .....	39
5.3 Intégrations .....	41
5.4 Construction du système .....	42
5.5 Résolution du système .....	43
6 Le Traitement des singularités .....	43
<b>CHAPITRE 3 : GREEN-EXPERT, ARCHITECTURE ET FONCTIONNEMENT</b>	<b>45</b>
1 Introduction .....	45
2 La Base de Données et son gestionnaire .....	47
2.1 Généralités .....	47
2.2 Les propriétés .....	48
2.3 Les Éléments de Référence .....	51
2.4 La création et le déplacement des équations .....	54
2.5 Les opérateurs de Calculs à logique d'Éléments .....	57
2.6 Les opérateurs de Calculs d'Intégrales .....	61
2.7 Les opérateurs de traitement des intégrales .....	65
2.8 L'opérateur Construction de Système .....	66



3 Le Générateur de Formulations .....	67
3.1 Généralités .....	67
3.2 La description de l'inconnue et des propriétés .....	67
3.3 La description des discrétisations autorisées de l'inconnue .....	68
3.4 La description des intégrales. ....	69
3.5 Les systèmes de l'équation .....	70
3.6 La description matricielle des systèmes .....	71
3.7 Le Post-Processing .....	72
4 La Description des Discrétisations .....	72
4.1 Généralités .....	72
4.2 Description des Régions de la Discrétisation .....	74
4.3 Description des Sous-domaines .....	74
4.4 La description des transformations géométriques .....	74
5 La Définition d'un Problème IF .....	77
6 Le Solveur de Green-Expert : Mif .....	79

## **CHAPITRE 4 : VALIDATIONS** **83**

1 Introduction .....	83
2 Couplage EF-IF .....	83
2.1 Préliminaires .....	83
2.2 Techniques employées .....	85
3 Validation de Green-Expert .....	97
3.1 Préliminaires .....	97
3.2 De l'Équation de Laplace à l'équation de Darcy ...	97
3.3 Un exemple de résolution 3D .....	119

## **CONCLUSION** **121**

## **BIBIOGRAPHIE** **123**

## **ANNEXE A1** **128**

## **ANNEXE A2** **130**

## **ANNEXE A3** **132**

Le calcul scientifique fut à l'origine de l'informatique et, en ces temps que certains n'hésitent pas à appeler *informationnels*, il suscite toujours des recherches approfondies dans le domaine qu'il a contribué à créer mais aussi dans ceux dont il tire ses bases théoriques : les mathématiques et la physique. En effet, si l'ordinateur ne fut au début qu'un moyen d'accélérer les calculs que l'homme pouvait effectuer (computer), il est maintenant devenu l'outil indispensable que l'on connaît dans tous les secteurs de la recherche scientifique.

La simulation numérique se situe dans la droite lignée de cette période héroïque où des kilos de cartons perforés étaient nécessaires pour résoudre une équation différentielle destinée, par exemple, à un calcul balistique. En cinquante ans, nous sommes passés des équations différentielles aux équations aux dérivées partielles (EDP), mais surtout, aucun domaine n'est épargné. Les sciences humaines comme les sciences physiques ont besoin de modèles numériques pour comprendre, prévoir, corriger les phénomènes qui nous entourent, en d'autres termes, il s'agit pour l'homme d'avoir une représentation mathématique de la réalité. Le grand avantage de cette modélisation de notre perception de la réalité par rapport à une approche expérimentale, toujours indispensable pour valider les modèles, est bien sûr le contrôle que l'on a sur les paramètres de la modélisation.

Cette possibilité de faire varier les divers paramètres d'une modélisation fait de la simulation numérique un puissant outil de compréhension et d'optimisation des procédés industriels. C'est pourquoi tous les secteurs industriels, de l'automobile à l'aérospatiale, en passant par la chimie et la métallurgie, font maintenant appel à des logiciels de simulation numérique pour concevoir et/ou optimiser leurs produits.

Devant la diversité et la complexité des problèmes physiques rencontrés, les logiciels de simulation numérique deviennent de plus en plus pointus. Dans les domaines en constante évolution que sont l'élaboration de procédés industriels et la modélisation numérique, le risque de cette spécialisation est un manque d'évolutivité du logiciel. C'est pour répondre à ce besoin d'évolutivité que nous avons effectué notre recherche, dans le prolongement de celle débutée il y a une dizaine d'années.

La recherche précédente avait abouti à un logiciel de simulation numérique, Flux-Expert, dont l'originalité est sa capacité à résoudre la plupart des équations aux dérivées partielles de la physique des milieux continus à condition qu'ils rentrent dans un cadre conceptuel précis. Ce cadre contient une méthode numérique de résolution, la Méthode des Éléments Finis, mais

surtout une décomposition analytique de cette méthode. Un logiciel Générateur de Formulations permet à l'ingénieur, au physicien, de décrire sa propre formulation dans ce cadre et de l'appliquer à son problème.

L'exploitation commerciale de ce logiciel a permis de valider la démarche mais aussi de la confronter aux réalités industrielles. La Méthode des Éléments Finis a ses limites sur lesquelles nous reviendrons et il était nécessaire de se pencher sur une autre méthode numérique. Notre attention s'est portée sur la Méthode des Intégrales de Frontières, dont la complémentarité avec la Méthode des Éléments Finis nous permet d'envisager un élargissement des domaines d'application des simulations numériques possibles.

Le chapitre 1 de ce mémoire est consacré à une présentation générale de la Méthode des Intégrales de Frontières (MIF). Cette présentation s'attache à distinguer les aspects fondamentaux des aspects complémentaires dans le double but d'une démarche didactique et d'une préparation à une décomposition analytique de la méthode.

Le chapitre 2 présente les principes de base de notre démarche dont le but est d'offrir à l'ingénieur un outil évolutif de conception de problèmes résolubles par la MIF. Cela est replacé dans le contexte des solutions actuellement disponibles sur le marché. Ce chapitre contient une analyse méthodologique de la définition et de la résolution d'un problème par la MIF.

Le chapitre 3 comprend la description du logiciel issu de notre travail : Green-Expert. Il en montre l'architecture et le fonctionnement.

Enfin, le chapitre 4 est consacré à la validation de notre travail sur des exemples concrets. Deux types de validations sont envisagés. Les premières concernent le couplage entre les Méthodes Intégrales de Frontières et Éléments Finis et les secondes montrent des exemples complets de modélisations réalisées avec Green-Expert, uniquement basées sur la Méthode des Intégrales de Frontières.

## 1 Introduction

---

La méthode des intégrales de frontières (MIF) est une méthode numérique qui a une double ascendance : les mathématiques liées aux méthodes intégrales et l'évolution des techniques numériques.

Elle puise en effet ses principes de base dans l'imposant travail accompli par les mathématiciens depuis le début du 19<sup>ème</sup> siècle dans le domaine des problèmes à valeurs limites, "Boundary value problems", ([50:STACKGOLD]). On cherche en effet depuis longtemps les solutions d'équations différentielles ou aux dérivées partielles dans des domaines de formes diverses et avec des conditions aux limites données.

Avant l'avènement du calcul numérique, on devait trouver une solution analytique exacte (ou éventuellement approchée à l'aide de séries infinies tronquées) au problème. Il a donc fallu rechercher des formulations mathématiques facilitant l'obtention de ces solutions. Les fonctions de Green ([48:ROACH]) ont été, et restent, des outils importants pour la résolution analytique des problèmes à valeurs limites et elles sont, comme on va le voir, à la base de la MIF.

La seconde source de la méthode provient du développement même des méthodes numériques et plus particulièrement de son aînée : la méthode des éléments finis (MEF). Le développement d'outils numériques nécessaires à cette dernière et la puissance grandissante des ordinateurs a permis d'envisager la résolution des systèmes produits par les méthodes intégrales qui recèlent des difficultés non négligeables tels que : système plein, singularités, mauvais conditionnement, etc....

Historiquement, les premières approches véritablement numériques des problèmes intégraux datent du début des années 60 ([31:JASWON], [51:SYMM]). La méthode est alors dite "indirecte", c'est à dire que les inconnues du problème sont des sources fictives qui permettent dans un deuxième temps de remonter à la variable d'état. Bien que les méthodes indirectes soient encore utilisées, nous nous intéresserons par la suite principalement aux méthodes directes qui calculent les variables d'état et présentent l'avantage d'avoir une plus grande généralité, ce qui est indispensable pour notre travail.

Une des premières formulations directes date de 1968 [14:CRUSE], mais la MIF n'existait pas comme méthode reconnue au même titre que la Méthode des Éléments Finis (MEF) qui était alors déjà bien répandue. C'est une équipe de l'université de Southampton qui a contribué à la reconnaissance de la méthode qui lui a d'ailleurs donné son nom ([6:BREBBIA]) : Boundary Element Method.

## 2 Aspects fondamentaux

### 2.1 Préliminaires

Il n'est pas dans notre intention de faire ici une présentation mathématique exhaustive et rigoureuse des fondements de la méthode, nous n'en avons point la place et cela nous mènerait hors de notre propos. Aussi, le lecteur mathématicien voudra bien nous excuser des raccourcis un peu rapides que nous effectuerons. Ils ont pour but de fournir les bases suffisantes à une compréhension globale du fonctionnement de la méthode. On se reportera aux références citées dans la bibliographie de ce chapitre pour une étude plus approfondie.

### 2.2 Position du problème

Nous nous intéresserons à la résolution d'un problème de la physique des milieux continus défini par un ensemble d'équations aux dérivées partielles vérifiées en tout point d'un espace  $\Omega$  de  $\mathbb{R}^n$  ( $n$  variant de 1 à 4; pour  $n=1$  c'est le champs des équations différentielles,  $n$  jusqu'à 4 pour les problèmes avec la variable temporelle) de frontière  $\partial\Omega$ . Ce problème admet des conditions aux limites (CAL) spatiales sur  $\partial\Omega$ , et des conditions aux limites temporelles (nous nous limiterons au cas des problèmes à conditions initiales) pour les variables d'états du système. On formalise ceci à l'aide de :

$$\left. \begin{array}{l} L(u) = f \quad \text{dans } \Omega \\ u \text{ vérifiant les CAL spatio-temporelles prescrites} \end{array} \right\} \quad (1.1)$$

où :  $L$  est un opérateur différentiel,  
 $u$  est une inconnue vectorielle composée des variables d'état du problème,  
 $f$  est une fonction quelconque, c'est le terme source.

En théorie, pour résoudre ce problème, on se place dans un espace fonctionnel  $F$  sur  $\Omega$  muni d'un produit scalaire noté  $\langle \cdot \rangle_\Omega$  et on cherche  $u$  tel que :

$$\left. \begin{array}{l} \forall w \in F, \langle L(u), w \rangle_\Omega = \langle f, w \rangle_\Omega \\ u \text{ vérifiant les CAL spatio-temporelles prescrites.} \end{array} \right\} \quad (1.2)$$

Pour être plus concret, l'espace fonctionnel habituel est  $L^2(\Omega)$ , espace des fonctions de carré sommable (c'est à dire intégrable au sens de Lebesgue) et son produit scalaire est alors défini par :

$$\langle u, w \rangle_\Omega = \int_{\Omega} u w \, d\Omega$$

L'expression (1.2) est une forme intégrale de (1.1). En fait, dans la plupart des méthodes numériques, et en tout cas dans les plus communément employées (Éléments finis, Volumes

finis, et Méthodes intégrales), on se ramène à une formulation intégrale du problème différentiel initial.

Les raisons qui expliquent cette mise en forme sont essentiellement mathématiques bien que les principes universels de la physique de conservation et/ou de minimisation d'énergie aient souvent des expressions intégrales analogues.

Comme on l'a suggéré dans l'écriture du problème, les bases mathématiques des méthodes numériques appartiennent au champ de l'analyse fonctionnelle. L'étude mathématique de la résolution des équations aux dérivées partielles (EDP) se situe dans des espaces munis de propriétés suffisamment régulières pour que les démonstrations de l'existence et/ou l'unicité des solutions soient possibles. Ce sont généralement des espaces de Hilbert. Si l'on se place dans un ouvert  $\Omega$  de  $\mathbb{R}^n$ , le produit scalaire dont on les munit généralement est :

$$\int_{\Omega} uv \, d\Omega$$

pour  $(u,v) \in H \times H$ ,  $H$  étant un espace de Hilbert.

Ce produit scalaire peut prendre des formes différentes, par exemple pour l'adapter au problème à résoudre [28:HRMADKA], mais il comporte toujours une intégration des fonctions. Dans l'espace fonctionnel  $H$ , la formule (1.2) est équivalente à (1.1) par définition du produit scalaire.

*Remarque* : L'équivalence entre (1.1) et (1.2) n'est qu'une version moderne du Lemme des variations qui stipule que si l'intégrale sur  $\Omega$  du produit d'une fonction  $f$  par n'importe quelle autre fonction est nulle, alors la fonction  $f$  est nulle en tout point de  $\Omega$ . Or, ce lemme est la clef de voûte du Calcul Variationnel qui a servi de base théorique à la méthode des éléments finis. De plus, rappelons que la résolution d'un problème variationnel passe classiquement par une intégration par partie -que l'on pourrait appeler, selon notre formation, principe de réciprocité ou théorème des travaux virtuels- qui permet d'aboutir à une formulation faible. Et c'est effectivement de cette façon que l'on va traiter le problème tel qu'il est posé en (1.2).

### 2.3 Formulation Faible

Après application de ce principe de réciprocité, le terme de gauche du problème (1.2) devient :

$$\langle L(u), w \rangle_{\Omega} = \langle u, L^*(w) \rangle_{\Omega} + \langle S^*(w), G(u) \rangle_{\partial\Omega} + \langle G^*(w), S(u) \rangle_{\partial\Omega} \quad (1.3)$$

où  $L^*$  est l'opérateur adjoint de  $L$ ,

$$C(u, w) = \langle S^*(w), G(u) \rangle_{\partial\Omega} + \langle G^*(w), S(u) \rangle_{\partial\Omega}$$

est la forme concomitante associée à  $L$ .



La notation  $\langle \cdot \cdot \rangle_{\partial\Omega}$  est la restriction du produit scalaire à la frontière  $\partial\Omega$  de  $\Omega$ . En d'autres termes, ce sont des intégrales de frontières. La formule (1.3) est parfois appelée dans la littérature "Formule de Green Généralisée" car on peut en déduire la célèbre troisième identité de Green pour l'opérateur Laplacien  $\Delta$ , comme on le verra plus loin.

En reportant (1.3) dans (1.2) le problème devient :

$$\left. \begin{aligned} \forall w \in F, \quad & \langle u, L^*(w) \rangle_{\Omega} + \langle S^*(w), G(u) \rangle_{\partial\Omega} + \langle G^*(w), S(u) \rangle_{\partial\Omega} = \langle f, w \rangle_{\Omega} \\ & G(u) = g \\ & S(u) = s \end{aligned} \right\} (1.4)$$

où  $u$  doit vérifier les conditions initiales prescrites

Les conditions aux limites spatiales ont été formalisées par deux opérateurs différentiels  $G$  et  $S$ . D'une façon similaire à  $L$ ,  $S^*$  et  $G^*$  sont les opérateurs adjoints de  $S$  et  $G$ .

La fonction  $S(u)$  est appelée condition aux limites imposée (*essential boundary condition*); typiquement, pour l'opérateur Laplacien  $\Delta$ ,  $S(u) = u$ , c'est à dire  $S = ()$  (Opérateur identité), on retrouve les conditions de Dirichlet.

La fonction  $G(u)$  est appelée condition aux limites naturelle (*natural boundary condition*); typiquement, pour l'opérateur Laplacien  $\Delta$ ,  $G(u) = \frac{\partial u}{\partial n}$ , c'est à dire  $G = \frac{\partial}{\partial n}$ , on retrouve les conditions de Neumann.

$g$  et  $s$  sont des fonctions quelconques.

L'expression (1.4) est une formulation faible de (1.2). On dit aussi que si  $u$  vérifie (1.4), c'est une solution faible de (1.2). Les avantages des formulations faibles sont de diverses natures. Cela permet de chercher des solutions dans un espace où les conditions de régularité sont moins strictes que dans la formulation initiale puisque l'intégration par partie a "transféré" les dérivées sur les fonctions  $w$  (que l'on appelle fonctions tests).

D'autre part, les mathématiciens ont produit un travail important sur l'existence et l'unicité des solutions faibles de plusieurs familles d'opérateurs. Pour ne citer que le plus célèbre d'entre eux, le théorème de Lax-Milgram démontre l'existence et l'unicité d'une solution faible pour les opérateurs elliptiques. Ces derniers couvrent une partie non négligeable de la physique puisque l'opérateur de Laplace est le modèle des opérateurs elliptiques.

Enfin, avantage pratique, la formulation faible fait apparaître des intégrales sur la frontière indispensables bien sûr pour la MIF, mais aussi pour la MEF car cela permet d'appliquer toutes sortes de CAL parfois plus "physiques" que des simples CAL de Dirichlet.

Les formulations faibles des Équations aux Dérivées Partielles sont les points de départ théoriques des principales méthodes numériques.

Il s'agit après cela de passer d'un espace fonctionnel de dimension infinie à un espace de dimension finie afin d'envisager un traitement numérique. On utilise pour cela la notion de discrétisation sur laquelle on reviendra plus en détail dans le chapitre suivant. Brièvement, il s'agit de découper le domaine de résolution en *éléments*. Ces éléments forment une discrétisation du domaine, ils contiennent un nombre fini de points qui vont servir de support aux fonctions utilisées dans cet espace de dimension finie.

#### 2.4 La méthode des résidus pondérés

Dans cet espace de dimension finie, on va choisir une base de fonctions  $\{ \psi_i \}$  (ou une famille de fonctions linéairement indépendantes mais on continuera par la suite à employer abusivement le terme "base" pour les deux) sur laquelle on va projeter les fonctions tests à l'aide de notre produit scalaire et une autre base  $\{ \alpha_j \}$  sur laquelle on va projeter l'inconnue. La recherche de la solution sera donc en fait la résolution d'un système où les inconnues seront les coefficients (réels ou complexes) multiplicateurs des fonctions de la base  $\{ \alpha_j \}$  et caractérisant la solution.

L'obtention de ce système est simplement obtenu en écrivant la nullité d'une expression dans l'espace muni de ces bases.

Prenons par exemple le problème suivant :

$$\left. \begin{array}{l} \text{Soit } M \text{ un opérateur linéaire différentiel, } F \text{ un espace fonctionnel} \\ \Omega \text{ un ouvert de } \mathbb{R}^n, \text{ trouver } u \text{ tel que :} \\ \forall w \in F, \quad \langle M(u), w \rangle_{\Omega} = 0 \end{array} \right\} \quad (1.5)$$

Par souci de simplicité, on ne tient pas compte des conditions aux limites (ou l'on suppose qu'elles sont uniformément nulles) mais cela n'enlève rien à la généralité des principes utilisés.

Comme  $\{ \psi_i \}$  est une base, (1.5) est équivalent à écrire la nullité du produit scalaire avec toutes les fonctions de la base. En utilisant la linéarité de  $M$  et la décomposition de  $u$  sur  $\{ \alpha_j \}$ , (1.5) devient :

$$\forall i, \quad \sum_j \langle M(\alpha_j), \psi_i \rangle_{\Omega} u_j = 0 \quad (1.6)$$

$$\text{si} \quad u = \sum_j \alpha_j u_j \quad \text{avec } u_j \in \mathbb{R} \text{ ou } \mathbb{C}$$

On obtient un système matriciel à résoudre  $[A](U) = 0$  avec :  $A_{ij} = \langle M(\alpha_j), \psi_i \rangle_{\Omega}$   
et :  $U_i = u_i$



Cette technique de résolution numérique par projection des fonctions tests sur une famille de fonctions indépendantes s'appelle la *méthode des résidus pondérés* [20:FINLAYSON]. Le nom de la méthode s'explique par le fait que l'expression  $\langle M(\sum_j \alpha_j u_j), \psi_i \rangle$  est appelé le résidu car c'est cette quantité que l'on cherche à annuler à l'aide des coefficients inconnus  $u_i$ , car elle est bien sûr non nulle pour une répartition quelconque des coefficients. Dans le cadre de cette technique, les fonctions  $\{ \psi_i \}$  sont appelées fonctions de pondération. Quant aux fonctions  $\{ \alpha_j \}$ , ce sont les fonctions d'interpolation de la variable  $u$ . On reviendra en détail plus loin sur ces considérations numériques.

Du point de vue de la méthode des résidus pondérés, la différence entre les diverses méthodes numériques réside dans le choix des fonctions de pondération. Celles de la méthode des intégrales frontières sont les solutions fondamentales de l'équation traitée (autrement appelées fonctions de Green comme on va le voir par la suite).

## 2.5 Solution fondamentale

Rappelons la formulation faible du problème après intégration par partie. Pour simplifier, on prend  $f=0$  dans (1.4) et on oublie provisoirement les conditions aux limites, on doit alors trouver  $u$  tel que :

$$\forall w \in F, \langle u, L^*(w) \rangle_{\Omega} + \langle S^*(w), G(u) \rangle_{\partial\Omega} + \langle G^*(w), S(u) \rangle_{\partial\Omega} = 0 \quad (1.7)$$

On a vu que la résolution numérique d'un tel problème nous amenait à choisir deux familles de fonctions. D'autre part, on constate dans (1.7) que si l'on est capable d'exprimer d'une façon simple son premier terme  $\langle u, L^*(w) \rangle_{\Omega}$  en fonction de  $u$  et ne dépendant plus de  $\Omega$ , on aura une formulation ne comportant que des intégrales de frontières.

Or, par définition, la solution fondamentale  $S^f$  pour  $L^*$  de pôle  $a \in \mathbb{R}^n$  est telle que :

$$L^*(S^f) + \delta_a = 0 \quad (1.8)$$

où  $\delta_a$  est la "fonction"<sup>1</sup> de Dirac de pôle  $a$ . De plus, la fonction de Dirac est telle que :

$$\forall u \in F, \langle u, \delta_a \rangle_{\Omega} = u(a) \quad (u \text{ au point } a, \text{ que l'on notera pour simplifier } u_a)$$

On comprend alors que, si l'on prend comme famille de fonctions tests (ou fonctions de pondération), des solutions fondamentales avec des pôles répartis dans le domaine  $\Omega$ , on pourra exprimer l'inconnue  $u$  en ces pôles en fonction uniquement d'intégrales de frontières,

<sup>1</sup> La fonction de Dirac, bien connue des Physiciens, qui vaut  $+\infty$  à son pôle et 0 ailleurs devrait être appelée distribution de Dirac. En effet, une étude mathématique rigoureuse des solutions fondamentales devrait être faite dans le champ de la théorie des distributions.

d'où le nom de la méthode. Ainsi, si on note  $\delta_i$  la fonction de Dirac associé à un point  $a_i$  du domaine  $\Omega$  et  $S_i^f$  sa solution fondamentale associée pour l'opérateur  $L^*$ , (1.7) devient :

$$\forall i, u_i = \langle S^*(S_i^f), G(u) \rangle_{\partial\Omega} + \langle G^*(S_i^f), S(u) \rangle_{\partial\Omega} \quad (1.9)$$

où  $u_i$  est la valeur de  $u$  pour le point  $a_i$ .

On voit que, pour aboutir à un système d'équations ne faisant intervenir que des valeurs discrètes de  $u$  sur la frontière  $\partial\Omega$ , il faut que le point  $a_i$  appartienne à  $\partial\Omega$ . D'autre part, il faut *discrétiser*  $u$ , c'est à dire, comme on l'a vu précédemment, projeter  $u$  sur une famille de fonctions. On étudiera le processus de discrétisation en détail dans le chapitre 3.

Le fait que  $a_i$  appartienne à  $\partial\Omega$  introduit un coefficient, dépendant de la géométrie de la frontière et du comportement de la solution fondamentale près de son pôle, que l'on notera  $c_i$ . On peut se reporter au paragraphe 3 de ce chapitre pour le calcul de ce coefficient appelé coefficient de collocation (*free-term coefficient*) pour l'opérateur de Laplace. On obtient donc finalement :

$$\forall i, c_i u_i = \langle S^*(S_i^f), G(u) \rangle_{\partial\Omega} + \langle G^*(S_i^f), S(u) \rangle_{\partial\Omega} \quad (1.10)$$

L'expression (1.10) est la forme générique des problèmes homogènes traités à l'aide de la MIF. Si le second membre de (1.1),  $f$ , est non nul, le problème est non-homogène et il apparaît classiquement des intégrales sur le domaine  $\Omega$ . Il existe cependant des techniques pour traiter ces intégrales de domaine qui sont plus ou moins bien adaptées à l'implantation de la méthode, comme on le verra dans 5.2.

Si l'expression (1.10) est bien la formulation générale nécessaire à la résolution d'un problème par la MIF, il est important de remarquer que l'expression précédente (1.9), avant le passage à la limite, sert, quant à elle, à calculer l'inconnue dans tout le domaine. En effet, (1.9) donne l'expression de l'inconnue pour tout point du domaine  $\Omega$  sous la forme d'intégrales de frontières qui font intervenir les valeurs de l'inconnue issues de la résolution de (1.10). Les deux formules sont donc importantes si l'on désire calculer l'inconnue sur la totalité du domaine.

Bien que nous n'ayons pas considéré les conditions aux limites au début de ce paragraphe pour aboutir à (1.10), il est clair que l'on devra les faire intervenir lors de sa résolution numérique. En effet, la solution fondamentale, telle qu'elle est définie par (1.8), n'en dépend pas. Or, si la prise en compte des conditions aux limites est relativement aisée dans le cadre d'une résolution numérique, il est aussi possible de chercher une fonction qui vérifie (1.4), tout en tenant compte des conditions aux limites (CAL).

## 2.6 Fonction de Green

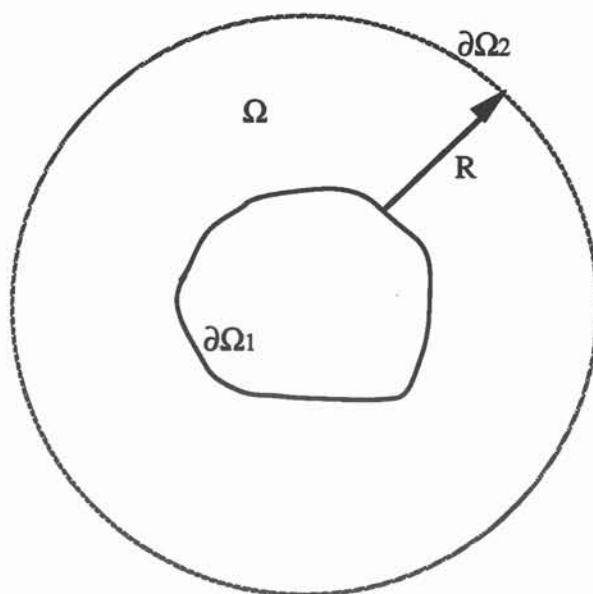
La fonction de Green est une notion antérieure à celle de la solution fondamentale, G. Green l'a utilisée bien avant la théorie des Distributions de L. Schwartz. Il s'en est servi pour déterminer les solutions analytiques de problèmes aux valeurs limites données. Les solutions étant analytiques, elles prenaient en compte les CAL. En fait, avec le recul que nous donne le siècle qui nous sépare de G. Green, *on peut définir une fonction de Green d'un problème donné comme étant une solution fondamentale qui vérifie un jeu de conditions aux limites pour ce problème.*

Comme la fonction de Green dépend aussi bien de la géométrie que des CAL, cela multiplie les possibilités théoriques de fonctions de Green et, dans la pratique de la résolution numérique, on préférera la généralité d'une solution fondamentale. De plus, il n'existe aucune méthode générale pour trouver la fonction de Green d'un problème particulier. On pourrait envisager un développement en série relativement aux fonctions propres de l'opérateur considéré ([48:ROACH], [50:STACKGOLD]) ou même une approximation numérique ([19:FENGSHENG]) mais nous pensons que cela enlève partiellement un des grands atouts de la méthode qui consiste à construire une solution approchée d'un problème à l'aide de solutions analytiques *exactes* du problème adjoint.

Dans un cas bien particulier, les solutions fondamentales vérifient cependant une partie des conditions aux limites : dans le cas où une des frontières du domaine de résolution s'étend à l'infini; en d'autres termes quand l'on cherche à résoudre un problème extérieur comme le schématise la figure 1.1.

Dans ce cas, la formule générique (1.10) des formulations intégrales de frontières doit être exprimée sur  $\partial\Omega_1$ , une frontière finie et sur  $\partial\Omega_2$ , la frontière qui tend vers l'infini. Or, il est clair que l'on ne peut discrétiser une frontière qui tend vers l'infini, on doit donc s'assurer que :

$$\lim_{R \rightarrow \infty} \langle S^*(S_i^f), G(u) \rangle_{\partial\Omega_2} + \langle G^*(S_i^f), S(u) \rangle_{\partial\Omega_2} = 0. \quad (1.11)$$



$$\Omega = \partial\Omega_1 + \partial\Omega_2$$

Quand  $R \rightarrow \infty$ ,  
 $\partial\Omega_2$  tend vers "l'infini"

### Un problème extérieur

Figure 1.1

Par exemple, en 2D, pour l'équation de Helmholtz, la solution fondamentale est

$$\frac{1}{4i} H_0^{(1)}(kr)$$

où  $H_0^{(1)}$  est la fonction de Hankel d'ordre 0 de première espèce,  $i$  le nombre imaginaire tel que  $i^2 = -1$ . Cette fonction est telle qu'elle vérifie la condition de régularité à l'infini, dite "condition de Sommerfeld" qui s'écrit, pour une fonction  $f$  :

$$\sqrt{\rho} \left( \frac{\partial f}{\partial \rho} - i k f \right) \rightarrow 0 \text{ quand } \rho \rightarrow \infty. \quad (1.12)$$

On peut montrer que cette condition est suffisante pour que la relation (1.11) soit vérifiée, la solution vérifiant aussi la condition de Sommerfeld (1.12).

On peut trouver des conditions de régularité à l'infini similaires pour la plupart des opérateurs, elles décrivent le comportement asymptotique des solutions de l'équation à résoudre et permettent de choisir la solution physiquement acceptable parmi toutes les solutions possibles.

### 3 L'exemple complet du Laplacien

L'exemple qui suit est à vocation essentiellement didactique. Il détaille les principales phases de la résolution d'un problème simple par la Méthode des Intégrales des Frontières. De plus, on y fera souvent référence, explicitement ou implicitement, pour illustrer pratiquement notre propos dans les chapitres suivants.

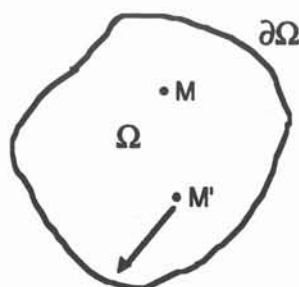
On veut résoudre dans un domaine  $\Omega$  l'équation de la chaleur :

$$\text{Div} ( -k_{th} \text{Grad}(T) ) = 0$$

où  $k_{th}$  est la conductivité thermique, et  
 $T$  la température, inconnue scalaire du problème.

Dans le cas où  $k_{th}$  est constant, on se ramène à  $\text{Div} ( \text{Grad}(T) ) = 0$ , soit l'équation de Laplace classique :

$$\Delta(T) = 0 \quad (\Delta : \text{Opérateur Laplacien})$$



Le point  $M'$  est le pôle de la solution fondamentale que l'on fait tendre vers la frontière.

La solution fondamentale de pôle  $M'$  de l'équation de Laplace en 2D est :

$$G_{M'}(M) = -\frac{1}{2\pi} \ln(MM'), \text{ on peut donc écrire :}$$

$$\forall M, M' \in \Omega, \Delta T(M) = 0 \text{ et } \Delta G_{M'}(M) + \delta(M') = 0. \quad (1.13)$$

Comme on l'a vu en 2.3, on applique la troisième identité de Green à  $G$  et  $T$  :

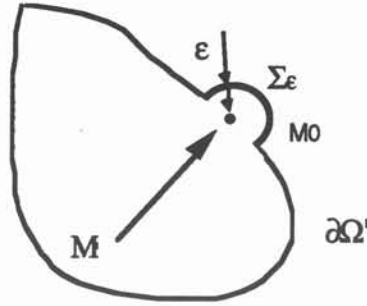
$$\int_{\Omega} ( \Delta T(M) G_{M'}(M) - \Delta G_{M'}(M) T(M) ) \, dM = \int_{\partial\Omega} ( G_{M'}(M) \frac{\partial T}{\partial n}(M) - T(M) \frac{\partial G_{M'}}{\partial n}(M) ) \, dM \quad (1.14)$$

où  $dM$  représente indifféremment un élément différentiel surfacique ou linéique.

Si l'on tient compte de (1.13) dans (1.14) on obtient :

$$\forall M' \in \Omega, T(M') = \int_{\partial\Omega} ( G_{M'}(M) \frac{\partial T}{\partial n}(M) - T(M) \frac{\partial G_{M'}}{\partial n}(M) ) \, dM \quad (1.15)$$

On fait tendre maintenant  $M'$  vers la frontière  $\partial\Omega$ . On effectue alors le schéma suivant où l'on entoure le point  $M_0$  où on veut faire tendre le point  $M'$  par un petit cercle de rayon  $\epsilon$ .



Alors, on doit effectuer le passage à la limite ( $\epsilon \rightarrow 0$ ), et l'on se retrouve avec une intégrale sur  $\Sigma_\epsilon$  et une autre sur  $\partial\Omega'$  (qui est égal à  $\partial\Omega - \Sigma_\epsilon$ ), c'est à dire :

$$T(M_0) = \lim_{M' \rightarrow M_0} T(M') = \lim_{\epsilon \rightarrow 0} \left\{ \int_{\partial\Omega} (G_{M'}(M) \frac{\partial T}{\partial n}(M) - T(M) \frac{\partial G_{M'}(M)}{\partial n}) dM + \int_{\Sigma_\epsilon} \left( -\frac{1}{2\pi} \text{Ln}(\epsilon) \frac{\partial T}{\partial n} + \frac{1}{2\pi\epsilon} T \right) \epsilon d\theta \right\} \quad (1.16)$$

La limite sur  $\partial\Omega'$  est par définition la valeur de l'intégrale sur  $\partial\Omega$  au sens de Cauchy soit :

$$\int_{\partial\Omega} (G_{M'}(M) \frac{\partial T}{\partial n}(M) - T(M) \frac{\partial G_{M'}(M)}{\partial n}) dM$$

Quant à la limite de l'intégrale sur  $\Sigma_\epsilon$ , comme les grandeurs  $T$  et  $\frac{\partial T}{\partial n}$  sont supposées finies, on a :

$$\lim_{\epsilon \rightarrow 0} \int_{\Sigma_\epsilon} \left( -\frac{1}{2\pi} \text{Ln}(\epsilon) \frac{\partial T}{\partial n} \right) \epsilon d\theta = 0 \quad (1.17)$$

et, si l'on suppose la frontière suffisamment lisse (en fait sans angles !! <sup>2</sup>)

$$\lim_{\epsilon \rightarrow 0} \int_{\Sigma_\epsilon} \left( +\frac{1}{2\pi} \frac{1}{\epsilon} T \right) \epsilon d\theta = \frac{T(M_0)}{2} \quad (1.18)$$

<sup>2</sup> En fait, si la frontière n'est pas lisse en  $M_0$ , il est simple de voir que la limite (1.18) est alors  $\frac{\alpha}{2\pi}$  où  $\alpha$  est le complémentaire de l'angle que fait la frontière en  $M_0$ . Ainsi, le coefficient de collocation  $c_i$  est en toute généralité, pour l'équation de Laplace,  $c_i = (1 - \frac{\alpha}{2\pi})$  c'est à dire, si on multiplie par  $2\pi$ , exactement l'angle que fait la frontière en  $M_0$ .

En faisant la somme de (1.17) et de (1.18), la limite de l'intégrale sur  $\Sigma\epsilon$  est donc  $\frac{T(M0)}{2}$ .

On en déduit finalement, en abandonnant par commodité la notation de l'intégrale au sens de Cauchy :

$$\forall M' \in \partial\Omega, \frac{1}{2} T(M') = \int_{\partial\Omega} (G_{M'}(M) \frac{\partial T}{\partial n}(M) - T(M) \frac{\partial G_{M'}(M)}{\partial n}) dM \quad (1.19)$$

Pour obtenir une formulation matricielle du problème, il ne nous reste plus qu'à discrétiser la frontière. On découpe la frontière en N morceaux  $\partial\Omega_i$ , i variant de 1 à N. Pour ne pas alourdir les notations, on considère T et  $\frac{\partial T}{\partial n}$  constants sur chacun des morceaux et l'on note  $T_i$  et  $\frac{\partial T_i}{\partial n}$  leurs valeurs respectives sur les morceaux  $\partial\Omega_i$ . (1.19) devient alors :

$$\forall j, \frac{1}{2} T_j = -\frac{1}{2\pi} \sum_{i=1}^N \frac{\partial T_i}{\partial n} \int_{\partial\Omega_i} \ln(M_i M_j) d(\partial\Omega_i) + \frac{1}{2\pi} \sum_{i=1}^N T_i \int_{\partial\Omega_i} \frac{\partial \ln(M_i M_j)}{\partial n} d(\partial\Omega_i) \quad (1.20)$$

où  $M_j$  est le point source et  
 $M_i$  le point courant sur le morceau  $\partial\Omega_i$   
 $d(\partial\Omega_i)$  est l'élément différentiel de  $\partial\Omega_i$

Soit la forme matricielle :

$$[H] [T] + [G] \left[ \frac{\partial T}{\partial n} \right] = 0 \quad (1.21)$$

avec :

$$H_{ij} = \frac{1}{2\pi} \int_{\partial\Omega_i} \frac{\partial \ln(M_i M_j)}{\partial n} d(\partial\Omega_i) - \frac{\delta_{ij}}{2}$$

$$G_{ij} = -\frac{1}{2\pi} \int_{\partial\Omega_i} \ln(M_i M_j) d(\partial\Omega_i).$$

Attention  $\delta_{ij}$ , n'est pas une fonction de Dirac mais le symbole de Kronecker qui vaut 1 quand  $i=j$  et 0 sinon.

Bien que l'équation de Laplace permet de résoudre un nombre non-négligeable de phénomènes physiques, ce n'est pas, loin s'en faut, la seule équation traitable par la Méthode des Intégrales de Frontières, comme nous allons le voir dans le paragraphe suivant.



## 4 Champs d'application de la MIF

Pour pouvoir résoudre un problème avec la méthode des intégrales de frontières, il faut pouvoir le mettre sous la forme :

$$\left. \begin{array}{l} L(u) = f \text{ dans } \Omega \\ \Omega \text{ un ouvert de } \mathbb{R}^n, L \text{ un opérateur différentiel} \\ \text{On connaît la solution fondamentale de l'opérateur adjoint } L^* \end{array} \right\} \quad (1.22)$$

Le champ est donc théoriquement aussi vaste que celui des éléments finis ou de toute autre méthode numérique. Il suffit en effet de reporter astucieusement les termes de l'opérateur de départ  $L$  en  $f$  pour que  $L^*$  devienne un opérateur "sympathique", c'est-à-dire dont on connaît la solution fondamentale pour le problème homogène. Il faut remarquer que certains opérateurs classiques de la physique sont auto-adjoints (c'est à dire que  $L^* = L$ ) et donc, il suffit alors de connaître la solution fondamentale du problème homogène associé.

Voici quatre équations simples, dont les opérateurs sont d'ailleurs auto-adjoints, pour lesquels on connaît la solution fondamentale :

*Nota* : Dans tout ce qui suit,  $r$  désigne la distance entre le "point source", pôle de la fonction de Dirac et le point d'application de la fonction. N'oublions pas qu'une solution fondamentale, tout comme une fonction de Green, est une fonction de deux points.

- *L'équation de Laplace* (elliptique)

Solutions fondamentales

$$\Delta(u) + \delta = 0$$

$$-\frac{1}{2\pi} \ln(r) \quad \text{en 2D}$$

$$\frac{1}{4\pi r} \quad \text{en 3D}$$

qui permet de résoudre tous les problèmes de diffusion linéaires homogènes de la physique des milieux continus en thermique, magnétostatique, électrostatique, mécanique du solide, mécanique des fluides, etc...

- *L'équation de Helmholtz* (hyperbolique)

Solutions fondamentales

$$\Delta(u) + \varepsilon \mu \frac{\partial^2 u}{\partial t^2} + \delta = 0$$

$$\frac{1}{4i} H_0^{(1)}(kr) \quad \text{en 2D}$$



qui devient en notation complexe :

$$\Delta(u) + k^2 u + \delta = 0$$

où  $H_0^{(1)}$  est la fonction de Hankel d'ordre 0 de première espèce,  $i$  le nombre imaginaire tel que  $i^2 = -1$

$$\frac{1}{4\pi r} \exp(-ikr) \quad \text{en 3D}$$

qui résout les problèmes de propagation dans un milieu homogène.

- L'équation dite "de la chaleur" (parabolique)

$$\Delta(u) - \frac{1}{k} \frac{\partial u}{\partial t} + \delta = 0$$

Solutions fondamentales

$$-\frac{1}{4\pi k t} \exp\left(-\frac{r^2}{4kt}\right) \quad \text{en 2D}$$

$$-\frac{1}{(4\pi k t)^{3/2}} \exp\left(-\frac{r^2}{4kt}\right) \quad \text{en 3D}$$

qui résout les problèmes de diffusion en transitoire dans les milieux homogènes. Dans ce dernier exemple, le traitement du temps est directement pris en compte dans la solution fondamentale (car la fonction de Dirac est, dans ce cas, spatio-temporelle). On peut noter ici que, tout comme pour les autres méthodes numériques, les problèmes transitoires peuvent être résolus classiquement avec un schéma aux différences finies, c'est à dire que l'on fait une décomposition de Taylor des dérivées temporelles et l'on se retrouve avec une inconnue discrétisée par rapport au temps à traiter à plusieurs instants et un facteur temporel multiplicatif  $\Delta t$ .

- L'équation de Navier

$$\sigma u_{j,kk} + \frac{\sigma}{1-2\nu} u_{k,kj} + \delta_j = 0$$

Solutions fondamentales

$$u_{ij} = \frac{(3-4\nu)\ln(1/r)\delta_{ij} + r_{,i}r_{,j}}{8\pi\sigma(1-\nu)} \quad \text{en 2D}$$

$$u_{ij} = \frac{(3-4\nu)(1/r)\delta_{ij} + r_{,i}r_{,j}}{16\pi\sigma(1-\nu)} \quad \text{en 3D}$$

qui résout les problèmes d'élasticité linéaire isotrope et homogène. Les notations sont celles utilisées classiquement dans le cadre de la mécanique pour alléger les écritures, c'est à dire les notations dites d'Einstein. On a ainsi :

$$r_{,i} = \frac{\partial r}{\partial x_i} \quad \text{et} \quad u_{k,kj} = \sum_k \frac{\partial u_k}{\partial x_k \partial x_j}$$

car la sommation est implicite si deux indices sont identiques. D'autre part,  $\sigma$  représente le module de Young,  $\nu$  le coefficient de Poisson et  $\delta_{ij}$  le symbole de Kronecker, à ne pas confondre avec  $\delta_j$  qui est la  $j^{\text{ème}}$  composante d'une fonction de Dirac vectorielle.

Ce sont bien sûr quatre exemples parmi les plus connus, ils sont tirés de [56:WALKER] qui en donne d'autres ainsi qu'une brève description de techniques d'obtention de solutions fondamentales. Par ailleurs, la plupart des références citées dans la bibliographie comportent explicitement la solution fondamentale (ou la fonction de Green) pour les formulations décrites. Elles recouvrent tous les champs de la physique des milieux continus : Thermique, Electromagnétisme, Mécanique des fluides et des solides.

Il est important de remarquer que toutes ces fonctions de Green sont connues pour des équations linéaires, sans terme source, dans des milieux homogènes. Il est cependant possible de traiter des sources dans des milieux non homogènes avec la MIF et, dans l'optique que nous nous sommes fixés, il est important de prendre en compte ces possibilités.

## **5 Aspects complémentaires**

---

### *5.1 Hétérogénéité du milieu traité*

Comme on l'a vu dans les aspects fondamentaux, une formulation IF classique s'applique naturellement sur un milieu homogène, c'est à dire où les propriétés physiques sont constantes. Si le milieu est hétérogène, il y a deux cas bien distincts :

- Le milieu est homogène par morceau : Dans ce cas, la procédure classique ([4:BREBBIA]) consiste à construire les systèmes IF normalement et séparément sur chacun des morceaux, que l'on appellera dorénavant "sous-domaine". La résolution globale sur l'ensemble des sous-domaines (Id. le domaine) est ensuite possible en assemblant dans le système numérique les relations de continuités entre les sous-domaines pour toutes les composantes de l'inconnue. Par la suite, on appellera l'interface entre deux sous-domaines classiques, un sous-domaine interfacial.

Cette technique de construction séparée est largement la plus répandue et permet la modélisation de phénomènes pour lesquels la relation interfaciale n'est pas forcément simple, comme par exemple la solidification dans [53:TONG]. Cependant, ce n'est pas la seule technique possible et certains auteurs préconisent l'utilisation de formulations qui prennent directement en compte les interfaces dans l'écriture des intégrales qui les caractérisent ([60:DONG],[40:MEDINA]). Cela peut notamment être utile lors de la résolution de problèmes multi-couches bien que ce cas est parfois assimilable au deuxième cas de l'hétérogénéité.

- Le milieu est complètement hétérogène : Dans ce cas, il n'est pas envisageable de découper le domaine en sous-domaines où les propriétés seraient constantes.

Il se peut alors qu'il existe une fonction de Green qui prenne en compte l'hétérogénéité directement dans sa formule comme c'est par exemple le cas pour l'anisotropie dans l'équation de Laplace. En effet l'équation :

$$k_x \frac{\partial^2 u}{\partial x^2} + k_y \frac{\partial^2 u}{\partial y^2} = 0$$

admet pour fonction de Green  $u = \frac{1}{\sqrt{k_x k_y}} \frac{1}{2\pi} \text{Ln}\left(\frac{1}{r}\right)$  où  $r = \sqrt{\frac{1}{k_x} r_x^2 + \frac{1}{k_y} r_y^2}$

Elle est par exemple utilisée dans [9:BRUCH] pour des calculs d'infiltration dans des sols anisotropes.

Mais lorsque qu'il n'existe pas de fonction de Green qui prenne en compte analytiquement le comportement hétérogène, on se retrouve alors soit dans le cas de termes sources additionnels par rapport à la formulation homogène (comme par exemple en thermoélasticité), soit dans le cas d'un traitement similaire à une non-linéarité, deux aspects que nous traitons dans les paragraphes suivants.

## 5.2 Termes sources

Si l'on veut résoudre un problème comportant des termes sources, la formulation IF comprend alors des intégrales sur le domaine (Cf (1.4) on doit calculer  $\langle f, w \rangle_\Omega$  où  $f$  est le terme source), au lieu de ne comporter que des intégrales sur la frontière. Ces intégrales de domaine font bien sûr perdre un peu l'avantage gagné par la méthode mais il existe des techniques efficaces permettant d'envisager la prise en compte de ces intégrales.

La première de ces techniques consiste tout simplement à calculer effectivement ces intégrales de domaine comme elles apparaissent dans la formulation. C'est sans doute la technique la plus générale mais aussi celle qui est la moins adaptée à la méthode puisqu'elle impose un maillage du domaine intérieur pour pouvoir calculer ses intégrales. Les techniques d'intégration de ces intégrales de domaine couvrent un champ assez large qui part de simples intégrations analytiques jusqu'à des techniques sophistiquées d'intégrations mélangeant analytique et numérique. On peut cependant remarquer que ces intégrations de domaine permettent de traiter le terme source sans rajouter d'inconnues au système à résoudre.

Pour pallier l'inconvénient d'avoir besoin d'un maillage intérieur, on dispose principalement de deux moyens :

- La transformations des intégrales de domaines en intégrales de frontières : cela nécessite un comportement particulier du terme source. Concrètement, tous les exemples que nous avons trouvés dans la littérature s'inspirent de [47:RIZZO]. Le terme source étant harmonique, c'est à

dire  $\Delta f = 0$ , on se sert alors d'une fonction auxiliaire  $v$ , telle que, si  $w$  est la fonction de Green de la formulation homogène, on ait  $\Delta v = w$ .

L'identité de Green permet alors d'écrire :

$$\langle f, w \rangle_{\Omega} = \langle f, \Delta v \rangle_{\Omega} = \langle f, \frac{\partial v}{\partial n} \rangle_{\partial\Omega} - \langle v, \frac{\partial f}{\partial n} \rangle_{\partial\Omega}$$

c'est à dire qu'il ne reste plus que des intégrales de frontières.

• La recherche de solutions particulières : on part ici de la formulation initiale du problème (1), soit  $L(u) = f$ , où  $L$  est un opérateur linéaire (à ne pas confondre avec la linéarité des propriétés physiques). En se servant de la linéarité de l'opérateur  $L$ , on va chercher la solution sous la forme d'une somme, soit :

$$u = u_h + u_p \quad \text{où}$$

$$u_h \text{ est la solution générale de l'équation homogène } L(u) = 0 \quad \text{et}$$

$$u_p \text{ est une solution particulière de } L(u) = f$$

Pour  $u_h$ , on se ramène donc au cas classique sans terme source. Il reste donc à trouver une solution particulière de  $L(u) = f$ . Deux cas se présentent encore, soit on connaît une solution analytique particulière et le problème est réglé, soit on n'en connaît pas et il faut la calculer numériquement. Dans ce dernier cas, il existe un certain nombre de techniques ([43:PARTRIDGE], [26:HENRY], [1:ALESSANDRI], [52:TANG]) qui ont toutes en commun de rechercher les coefficients inconnus de l'approximation de  $u_h$  par une série de fonctions :

$$u_h = \sum_{i=1}^N \eta_i F_i$$

La différence entre les techniques existantes réside essentiellement dans le choix des fonctions  $F_i$ . On retiendra cependant particulièrement la technique proposée initialement dans [42:NARDINI] qui s'appela par la suite la "Dual Reciprocity Method-DRM" ([43:PARTRIDGE]) et qui consiste à écrire :

$$\Delta \hat{u}_i = F_i \quad \text{où } i \text{ réfère un nœud de la discrétisation.}$$

On peut alors appliquer l'identité de Green pour la série ainsi obtenue. L'intérêt de la méthode est d'obtenir une formulation, pour l'obtention de la solution particulière, qui ne contient que des intégrales de frontières dont les formules sont similaires à celles obtenues lors d'un traitement classique de la méthode. L'implantation de cette méthode serait donc théoriquement relativement plus simple qu'une autre dans un code IF. Il faut cependant remarquer que, d'une part, elle exige une boucle supplémentaire, due à l'introduction de la série, par rapport à une formulation classique et que, d'autre part, l'ajout de nœuds intérieurs s'avère indispensable pour l'approximation d'un terme source non-trivial. Par contre la méthode DRM, mais très certainement aussi la plupart des méthodes basées sur une expansion

de la solution particulière en série de "fonctions de points", présente l'avantage non-négligeable d'être compatible avec une formulation non-linéaire.

### 5.3 Non-linéarités

Le traitement de problèmes non-linéaires s'avère indispensable pour une modélisation, que l'on veut de plus en plus proche de la réalité, des procédés industriels. Or, il est clair que la MIF n'est pas naturellement disposée au traitement des non-linéarités car, contrairement à la MEF, ce n'est pas une méthode locale, et que les phénomènes non-linéaires sont intrinsèquement locaux.

Néanmoins cet inconvénient doit être relativisé. En effet, de nombreux auteurs ont résolu certains problèmes non-linéaires en utilisant des techniques diverses. Une des techniques fréquemment rencontrées est celle de la transformation de Kirchoff appliquée aux problèmes elliptiques ([30:INGHAM], [8:BREBBIA]). Dans [8:BREBBIA], la technique est associée à une méthode DRM pour résoudre des problèmes thermiques non-linéaires transitoires.

Nous décrivons ci-dessous brièvement le principe de la transformation de Kirchoff. L'équation non-linéaire de départ est :

$$\text{Div} ( f(u) \text{ Grad}(u) ) = 0$$

à laquelle on applique la transformation de Kirchoff :

$$\text{Grad}(v) = f(u) \text{ Grad}(u)$$

L'équation de départ est linéarisée et devient alors :

$$\Delta v = 0 \quad \text{dont on sait dériver une formulation IF.}$$

La linéarisation du problème de départ ne doit pas faire oublier qu'il était effectivement non-linéaire. Ainsi, la résolution d'un tel problème nécessitera l'application d'un schéma itératif classique, de la même manière que pour une autre méthode numérique.

Cette explication montre bien la principale difficulté du traitement de problèmes non-linéaires par une formulation IF : *la linéarisation*.

En effet, toute résolution d'un problème non-linéaire passe par cette étape indispensable qu'est la linéarisation de la formulation, puisque, finalement, il faudra bien résoudre un système matriciel qui contient des coefficients constants.

Dans le cas d'une méthode locale comme la MEF, cette étape de linéarisation initiale de la formulation n'est pas obligatoire car rien n'empêche une formulation Éléments finis, discrétisée avec la méthode des résidus pondérés par exemple, d'être non-linéaire. Or, ce n'est pas le cas d'une formulation Intégrales de Frontières qui exige une formulation linéarisée a priori pour

supporter l'application de la formule de Green Généralisée vue en 2.3 (ou principe de réciprocité).

Quand il n'existe pas de changement de variable, comme pour la transformation de Kirchoff, qui permette d'aboutir à une formulation linéarisée, il est possible, la plupart du temps, d'isoler les termes non-linéaires de la formulation et de les traiter comme des termes sources. Cette technique est notamment utilisée en mécanique des fluides ([4:BREBBIA], [2:BANERJEE], [58:WU]).

Le rejet des termes non-linéaires en terme source n'est pas la seule solution. Comme cela est exposé dans [55:TOSAKA], on peut aussi découper suffisamment le domaine pour envisager une linéarisation simple de l'opérateur à traiter qui serait valide dans chacun des sous-domaines obtenus.

Nous ne pouvons pas ici effectuer un exposé exhaustif sur le vaste problème posé par le traitement des non-linéarités dans le cadre de la MIF. Il faut souligner qu'il est fréquent, dans une modélisation physique, d'avoir des comportements non-linéaires localisés dans des matériaux précis. Si la formulation le permet, il peut être alors pratique de pouvoir traiter ces non-linéarités par la MIF. Cependant, il est clair qu'un couplage entre la MEF (pour des non-linéarités compliquées) et la MIF (pour des domaines linéaires, infinis ou à surface libre par exemple) s'avère être un moyen puissant de pallier les faiblesses respectives des deux méthodes. Nous reviendrons en §4 sur l'intérêt du couplage entre ces deux méthodes.

#### *5.4 Les singularités*

Les formulations Intégrales de Frontières fournissent une large palette de ce que nous avons regroupé sous le terme générique de "singularités". On distingue deux grandes familles de singularités. Les singularités qui sont intrinsèques à la méthode, causées par l'utilisation de fonctions de Green qui comportent des noyaux singuliers, et celles qui sont intrinsèques à la physique.

Dans une modélisation numérique, les singularités physiques créent des lieux où la valeur d'une variable physique est soit indéterminée, soit multiple, soit infinie. La plupart du temps, dans une modélisation, et ceci quelle que soit la méthode numérique employée, on évite soigneusement ces lieux. En ce qui concerne la MIF, on trouve souvent la démarche inverse : peut-être la présence naturelle de noyaux singuliers dans les formulations, a-t-elle conduit de nombreux auteurs à s'intéresser directement à ces singularités physiques.



En fait, mis à part le cas d'une source ponctuelle<sup>3</sup>, (puisque l'on reste dans le cadre de la physique des milieux continus), une grandeur, quelle qu'elle soit, ne peut, ni être indéterminée, ni prendre des valeurs infinies en un lieu donné si son voisinage immédiat (au sens mathématique du terme) ne comporte aucune discontinuité. Une discontinuité pouvant être d'origine géométrique (coin, arête, etc.), physique (interface avec un autre matériau), ou numérique (imposition d'une valeur différente).

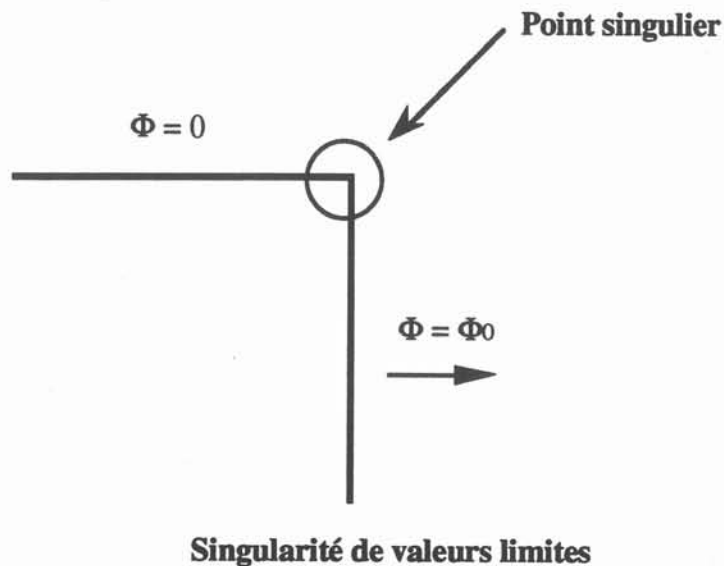


Figure 1.2

Si l'on pousse le raisonnement un peu plus loin, on se rend compte qu'une discontinuité géométrique entraîne une discontinuité numérique des variables en dépendant (comme la dérivée normale d'une grandeur) et qu'une discontinuité physique entraîne aussi une discontinuité des grandeurs concernées. *On se rend compte pratiquement qu'une singularité physique découle toujours d'une discontinuité de valeurs limites.*

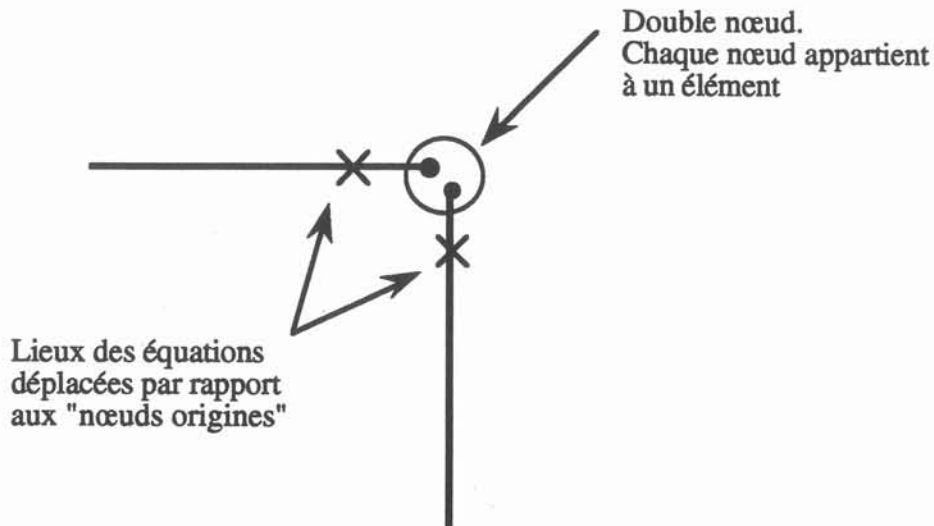
La figure 1.2 en montre un exemple dans le cas trivial d'une discontinuité de flux pour un problème potentiel.

Puisqu'il y a discontinuité, il est logique de penser qu'il faut pouvoir définir une grandeur plusieurs fois au même endroit. C'est effectivement une technique de multi-nœud qui fut préconisée assez tôt dans [13:CHAUDONNERET]. Le problème posé par l'adjonction de nœuds supplémentaires pour la prise en compte de singularités est le déficit en équations que

<sup>3</sup> La modélisation d'une source ponctuelle est d'ailleurs particulièrement bien adaptée à une formulation IF puisqu'elle se ramène souvent à l'ajout d'une distribution de Dirac à l'endroit de la source. Ainsi, dans la formule (1.10), de simples nombres réels caractéristiques des sources ponctuelles seraient ajoutés au second membre comme on peut le voir dans la référence [4:BREBBIA].

cela provoque. En effet, on a ajouté des inconnues mais pas d'équations. Il y a deux façons de résoudre ce problème : numériquement ou analytiquement.

Si l'on veut rajouter une équation numérique, le choix est limité à l'extérieur du domaine ([41:MITRA]) ou sur la frontière, mais "loin" du multi-nœud, l'équation étant déplacée par rapport à la singularité ([37:KRÄHENBÜHL]), cette dernière technique présentant l'indéniable avantage de laisser les équations sur la frontière comme le montre la figure 1.3.



**Déplacement d'équations près d'une singularité**

Figure 1.3

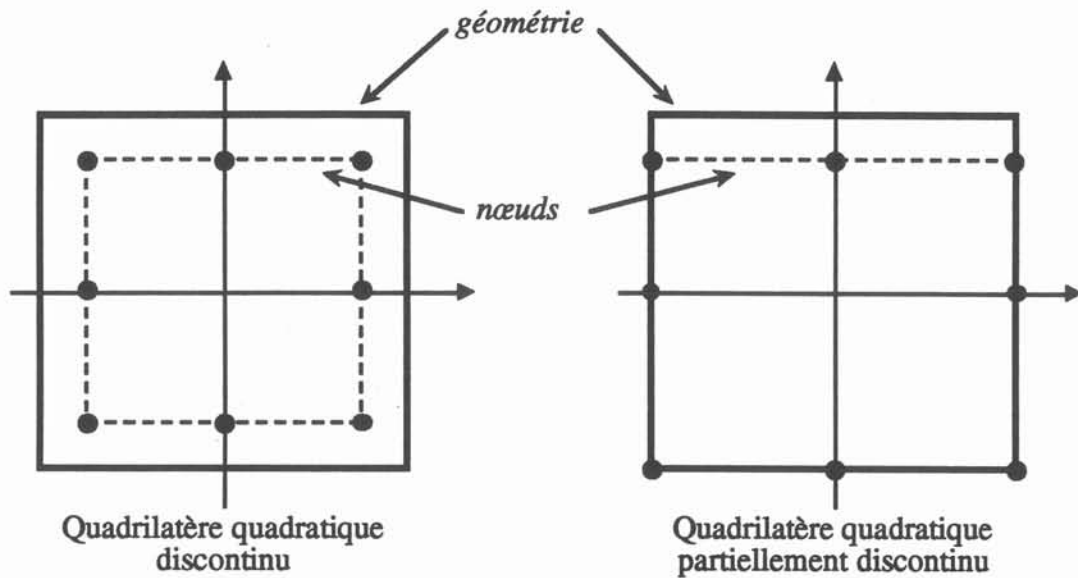
Il existe des cas où l'on connaît une relation analytique entre les nœuds du multi-nœud ([13:CHAUDONNERET], [36:KOST]), cela évite d'avoir à rajouter une équation purement numérique.

Enfin, il y a un certain nombre d'auteurs qui s'attachent à calculer numériquement le comportement analytique de la grandeur singulière à l'aide d'une approximation par des séries ([22:GRAY], [35:KELMANSON], [12:CANOT]). Il faut souligner une étude assez complète des points singuliers dans [12:CANOT] ainsi qu'une analyse mathématique ([23:GRISVARD]) qui traitent des singularités rencontrées dans un grand nombre de problèmes.

Toutes ces approches cherchent à évaluer au mieux la singularité. Pourtant, il existe bien des cas où il importe peu d'évaluer les inconnues près de cette singularité, il suffirait simplement de faire en sorte que sa présence ne provoque que des erreurs négligeables ailleurs. Les éléments discontinus [45:PATTERSON] et les éléments partiellement discontinus



[46:PATTERSON] répondent parfaitement à cette possibilité puisque les nœuds de ces éléments sont complètement (ou en partie) à l'intérieur de la géométrie de l'élément comme le montre la figure 1.4. Il est clair que ces éléments permettent d'éviter toutes discontinuités de valeurs limites. Cependant, ils ne dispensent pas d'un traitement spécifique pour la gestion des interfaces entre deux sous-domaines.



### Exemple d'éléments discontinus

Figure 1.4

Revenons sur les singularités intrinsèques à la méthode, c'est à dire le calcul des intégrales rendues singulières par le noyau de la fonction de Green. Il y a trois manières d'envisager le calcul de ces intégrales singulières : analytique, numérique, et semi-analytique (qui mélange techniques analytiques et techniques numériques).

Nous passerons rapidement sur l'intégration totalement analytique qui est bien sûr la meilleure mais n'est possible que dans certains cas. Les premières implémentations de la MIF, et encore beaucoup d'implémentations "locales" de la méthode dans les laboratoires, sont basées sur des éléments sur lesquels les valeurs sont supposées constantes (ordre 0) et permettent ainsi des intégrations analytiques. Cette approche qui peut paraître grossière permet cependant d'obtenir économiquement des résultats intéressants.

L'inconvénient majeur du traitement analytique est son manque de généralité et a conduit la majorité des auteurs à s'intéresser aux traitements numériques et semi-analytiques. Il serait compliqué de présenter exhaustivement les différentes solutions apportées d'autant plus qu'elles

varient souvent par des différences d'implémentation. Le principe qui semble s'imposer est celui d'un passage en coordonnées polaires avec origine au pôle. On associe à cette technique un changement d'élément de référence ([27:HONG-BAO LI], [59:WU ZHANG], [33:KATAYAMA]) ou un développement en série de Taylor de la fonction à intégrer ([25:HAYAMI], [24:GUIGGIANI], [15:CRUSE]) et cela permet d'appliquer des formules d'intégrations numériques de Gauss classiques ou adaptées. Dans les trois dernières références citées, on trouve des bibliographies assez complètes sur le sujet ainsi qu'un travail de généralisation remarquable.

Comme on le voit la prise en compte des singularités dans la MIF a conduit à un abondant travail dont nous avons donné un bref aperçu. Dans l'optique de notre travail, il est important de déduire de ces différents travaux, des règles simples qui induiront des structures logicielles qui ne seront pas incompatibles avec une implantation des solutions préconisées les plus générales.

Il y a en fait trois règles très simples qui, comme on le verra au §3.3, vont guider notre vision d'un traitement général de la MIF :

- On doit pouvoir créer des points géométriques où une grandeur pourra être définie plusieurs fois (concept du multi-nœud);
- On doit disposer de toute la latitude possible pour choisir les localisations des nœuds (endroits où les inconnues sont calculées) et des équations (endroit où sont situés les pôles des fonctions de Green);
- On doit dynamiser complètement la manière dont les intégrales peuvent être calculées, c'est à dire prévoir qu'une intégrale peut être calculée de plusieurs manières différentes et même faciliter la création de nouvelles techniques de calcul.

Comme on le voit, le champ d'application de la MIF est suffisamment vaste pour que l'on s'intéresse à la réalisation d'un système permettant de décrire interactivement une formulation et la faire résoudre par un solveur unique sur une géométrie quelconque, puisque tel est le principe du couple générateur - solveur généralisé.

### 1 Introduction

---

Le but de notre étude est de fournir à l'ingénieur, au physicien, un outil lui permettant de mettre en œuvre le plus simplement possible la simulation numérique de *son problème* de la physique des milieux continus. L'enjeu, d'énoncé simple, prend toute sa mesure quand on observe la complexité grandissante des phénomènes physiques simulés. On cherche en effet à s'approcher de plus en plus de la réalité physique des phénomènes étudiés jusque dans leurs aspects les plus singuliers. Pour faire face à cette évolution, les outils généraux sont souvent trop lourds et les outils spécifiques manquent de généralité. En effet, il ne suffit pas de savoir simuler un phénomène particulier de la physique, il faut pouvoir coupler divers phénomènes pour simuler des procédés industriels de plus en plus complexes.

Avant de présenter notre démarche pour répondre à ces besoins, nous allons préciser la définition de notre champ d'étude : la simulation numérique, puis nous envisagerons les diverses approches possibles pour répondre à la problématique présentée ci-dessus.

### 2 La simulation numérique

---

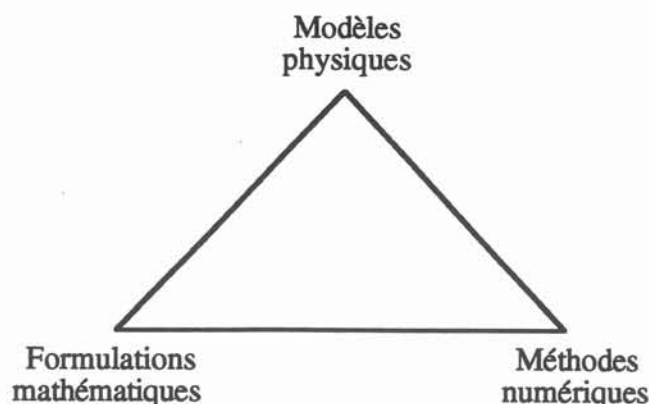
La simulation numérique se propose de calculer numériquement, c'est à dire pratiquement en un certain nombre de points d'une géométrie donnée, des grandeurs déterminées par des équations. On dit que ces équations sont résolues numériquement. Il est intéressant de remarquer, qu'avant l'avènement des ordinateurs, on cherchait plutôt à calculer les coefficients d'une série de fonction qui convergeait vers la solution analytique de l'équation traitée pour une géométrie donnée.

Nous nous plaçons dans le cadre de la physique des milieux continus, les équations auxquelles nous sommes confrontés sont donc des équations aux dérivées partielles, des équations intégrales, voire des équations intégral-différentielles qui mêlent les deux précédents types.

La simulation numérique offre un moyen moderne et économique de faire varier les paramètres d'un procédé industriel dont on veut simuler le fonctionnement. Mais avant de discuter des degrés de liberté dont dispose le physicien pour utiliser un logiciel de simulation, intéressons-nous plutôt à ceux dont nous disposons pour concevoir un tel logiciel.

La simulation numérique emprunte à la physique dont elle utilise les modèles de représentation de la réalité, aux mathématiques dont les équations sont la base, et enfin à l'informatique qui est le support des méthodes numériques employées. Ces trois domaines

forment le triptyque de toute démarche de résolution numérique d'un problème de la physique des milieux continus symbolisé par la figure 2.1.



*Le triptyque de la modélisation numérique*

Figure 2.1

Pour ces trois étapes, on dispose de degrés de liberté. Les choix peuvent être faits par le *l'auteur* du logiciel de simulation où reportés sur *l'utilisateur* du logiciel, ce qui en augmente les possibilités mais peuvent en rendre l'utilisation plus compliquée :

- Choix dans la modélisation physique : Il n'est pas toujours évident de déterminer à priori les équations de la physique qui permettent d'approcher la réalité du comportement d'un dispositif industriel. C'est un premier niveau de choix dans la modélisation physique qui est souvent négligé : plusieurs équations doivent pouvoir être traitées afin de permettre de choisir le modèle le plus convenable. Le deuxième niveau est celui des propriétés physiques dont la modélisation consiste à donner les lois de comportement en fonction de tous les paramètres du système. Cette modélisation est susceptible d'évoluer en fonction d'expériences, de nouvelles connaissances, etc. Il est même possible que cette modélisation soit d'un caractère hautement confidentiel pour l'utilisateur.

- Choix des formulations mathématiques : Il y a bien des manières de poser mathématiquement un problème à résoudre. La modélisation consiste ici à choisir une méthode qui nous permet de mettre en forme les équations de la physique (souvent des EDP) pour qu'elles soient traitables par les méthodes numériques choisies.

- Choix des méthodes numériques : Le traitement numérique d'une équation consiste à passer formellement du monde réel (continu), représenté souvent par des EDP, au monde numérique (discret), qui est celui des matrices et des vecteurs. Cela se passe en général en deux

étapes : La première consiste à passer des équations de départ à une formulation du problème caractéristique de la méthode employée (on parle ici du nom générique de la méthode qui désigne l'ensemble du processus, par exemple méthode des intégrales de frontières). C'est en quelque sorte une étape de préparation des équations. La deuxième étape consiste alors à *discrétiser* cette formulation, c'est à dire faire en sorte qu'elle soit résoluble par des méthodes numériques. Cela consiste généralement à choisir des espaces fonctionnels de projection de dimension finie.

Une fois le problème posé d'une manière discrète, il faut le mettre sous forme matricielle, calculer les coefficients des ces matrices, et enfin résoudre le système. Ces étapes numériques font appel à des techniques variées d'intégration numérique, d'assemblage, de stockage, de résolution, etc. La complexité grandissante des problèmes résolus est à mettre naturellement en parallèle avec l'évolution continue de ces techniques numériques.

Il y a donc trois grands niveaux d'intervention dans la mise en œuvre d'une modélisation et nous allons voir comment ils interviennent dans les diverses approches possibles.

### **3 Les diverses approches**

On peut distinguer trois grandes familles de solutions apportées au problème de la conception d'une simulation numérique d'un problème donné :

- **Les grands logiciels** : Ces logiciels ont pour vocation de répondre à un besoin précis. Ils se nomment NASTRAN, IDE-AS, SYSTUS, etc. Que leurs champs d'application soient larges ou précis, qu'ils traitent des phénomènes simples ou complexes, ils ont tous en commun le fait que leur potentiel d'adaptation par l'utilisateur soit très faible. Ce type de logiciel permet de répondre rapidement et souvent très efficacement à un problème à partir du moment où il rentre dans ses possibilités, mais seule l'équipe qui travaille sur le produit peut élargir significativement son champ d'application à tous les niveaux d'intervention répertoriés au chapitre précédent. Il est évident que ce modèle n'est pas à suivre pour le but que nous nous sommes fixés.

- **Les bibliothèques** : Qu'elles soient généralistes comme NAG, IMSL, ou plus spécialisées comme MODULEF, ELLPACK, les bibliothèques ont beaucoup de succès, et particulièrement dans les milieux universitaires, d'où la plupart sont issues. Elles offrent à leurs utilisateurs toute une palette de programmes qui va du module de lecture des données à leur visualisation 3D en passant par les méthodes numériques les plus éprouvées. Cela permet de ne pas reprogrammer ce qui a déjà été fait, et de se concentrer uniquement sur la partie originale de son travail. L'utilisateur dispose de toute la latitude voulue tant au niveau des méthodes



numériques qu'au niveau des formulations mathématiques, quitte à en programmer si elles n'existent pas dans la bibliothèque, sans pour autant bénéficier d'un soutien pour leur implantation. Cela exige de bonnes connaissances en informatique et aussi dans l'utilisation de la bibliothèque choisie et ramène bien sûr des contraintes de programmation dues à l'apport extérieur. C'est une solution adaptée à un utilisateur qui, non seulement a des connaissances solides en informatique, mais aussi qui aura (ou saura) décomposer analytiquement son problème en vue de son traitement informatique. Ce n'est pas toujours le cas du physicien qui dispose rarement d'une équipe d'informaticiens capables d'analyser son problème.

- Les systèmes évolués : Sous ce terme, nous désignons principalement les logiciels qui disposent de ce que l'on peut appeler un langage de macro-programmation. Ce langage est, la plupart du temps, interprété; il permet de décrire, d'une manière abstraite, le problème à résoudre. Le modèle le plus connu de ce type de logiciel est MATHEMATICA qui est très généraliste mais il existe aussi des études faites dans le domaine de la simulation numérique. [18:DYKSEN], [34:KEAVEY]. Il offre un niveau d'abstraction plus ou moins élevé suivant son degré de généralité : il est clair que plus il est spécialisé dans un domaine, plus il "comprendra" la terminologie du domaine et moins l'utilisateur aura d'efforts de programmation à faire. A l'inverse, un logiciel comme MATHEMATICA (qui inclut, dans sa dernière version, un module "Différences Finies" et qui probablement inclura par la suite des méthodes plus élaborées telles que "Volumes Finis" ou "Éléments Finis") nécessite un investissement personnel non négligeable si on veut l'utiliser dans des fonctionnalités évoluées. Ce type de logiciel peut voir son utilisation facilitée par des bibliothèques de modules macro-programmés qui, à l'instar des bibliothèques précédemment citées, permettent de se consacrer à son propre problème, et, comme ces dernières, admet un certain nombre de degrés de liberté au niveau des méthodes numériques et des formulations mathématiques. Les latitudes sont toutefois moins larges que pour l'emploi direct de bibliothèques puisqu'il l'on ne peut sortir du cadre formel imposé par le logiciel.

La figure 2.2 propose une grille de lecture de ces différentes solutions de mise en œuvre d'une modélisation numérique. On y a privilégié deux aspects : l'adéquation au problème posé et la simplicité<sup>1</sup> d'utilisation. Nous y avons représenté ce que nous pensons être le spectre moyen des solutions apportées à un problème donné.

---

<sup>1</sup> La simplicité est une notion éminemment subjective. Pour fixer les idées, considérons que tout effort pour adapter le formalisme du problème de l'utilisateur dans son langage au formalisme de la solution logicielle munie, elle aussi, d'un langage, tend à diminuer la simplicité de la solution.

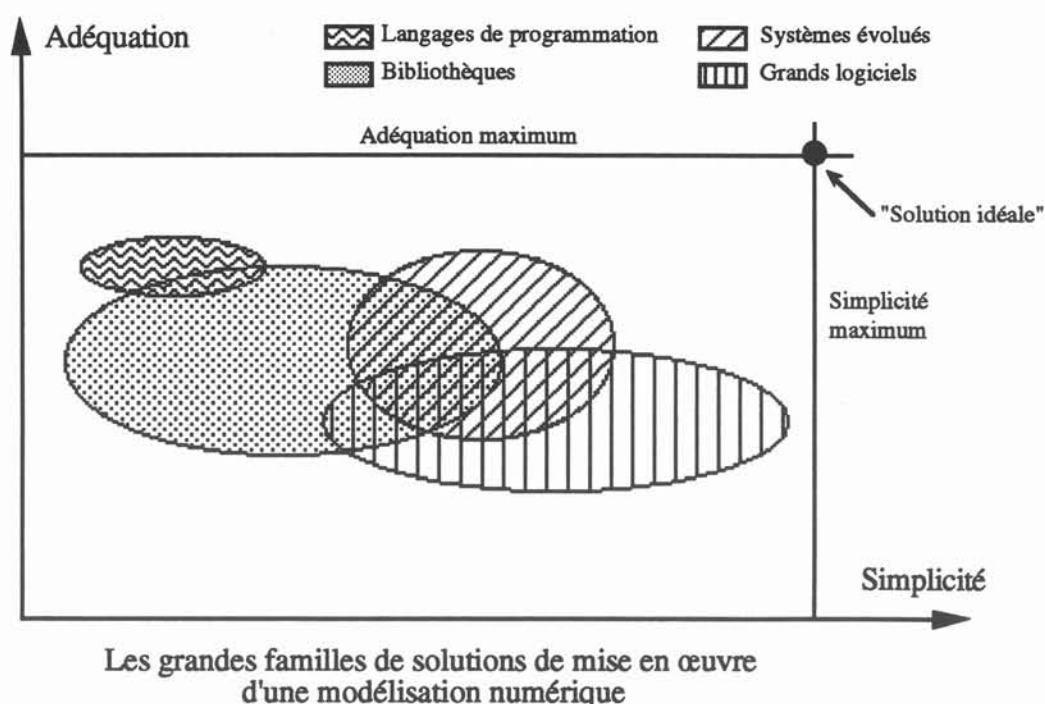


Figure 2.2

Dans l'espace adéquation-simplicité, une solution logicielle est représentée par une surface puisque son adéquation peut être agrandie par l'utilisateur, au détriment de sa simplicité. Cette figure ne représente en rien une réalité statistique et ne reflète que notre vision à un instant donné; nous espérons cependant qu'elle offre une image assez proche de la réalité. En plus des trois grandes familles décrites ci-dessus, nous y avons fait figurer les langages de programmation qui constituent bien sûr l'ensemble des solutions qui présentent la meilleure adéquation potentielle mais la plus mauvaise simplicité.

- Le cas des systèmes experts : Nous n'avons pas fait figurer le cas spécifique des systèmes experts (SE), bien que cela soit, à notre avis, la seule famille capable de fournir la "solution idéale" de simplicité et d'adéquation maximum, car il n'existe pas encore de véritable système expert significatif dans le domaine de la simulation numérique. Il existe des travaux effectués dans ce sens [38:LOWTHER], [21:FORD], [44:PESKIN], [49:ROZENBLIT] mais on en est encore aux prototypes. Comme on a pu le remarquer dans la description des diverses approches, il y a un champ libre que seuls les SE pourraient occuper : celui de la modélisation physique. Malheureusement, il n'existe pas de système expert capable, à partir d'une description purement physique d'un problème, de fournir une solution optimisée sur un espace discrétisé en ayant préalablement choisi le modèle physique le plus adapté, la formulation mathématique et les méthodes numériques capables de résoudre au mieux le problème. Cette solution idéale, qui se situerait à l'intersection des deux maxima de notre figure adéquation-simplicité, est sans aucun doute un des points de mire des années 90 de la recherche en

modélisation. Mais, avant d'arriver à un tel système, il faut pouvoir disposer d'outils de conception et de résolution qui couvrent le spectre le plus large possible de la physique des milieux continus tout en offrant le maximum de degrés de liberté aux niveaux des formulations mathématiques et des méthodes numériques.

C'est dans cette optique que se situe notre démarche.

#### **4 Principes du couple générateur d'équations - solveur généralisé**

---

Notre travail se situe dans le prolongement d'une recherche débutée il y a plus de quinze ans, les principes que nous allons énoncer ont donc déjà fait l'objet d'une thèse d'état [39:MASSÉ] et sont à l'origine d'un logiciel, FLUX-EXPERT, basé sur la méthode des éléments finis [17:DT2i], industrialisé depuis 1986. Cependant, toute l'analyse ayant été refaite dans le cadre d'une autre méthode numérique, les concepts fondamentaux de la démarche ont été revus et généralisés dans une optique multi-méthode.

L'originalité de la démarche réside dans la matérialisation, sous la forme d'un fichier, de la description formelle de toute équation (une formulation) associée à un Solveur capable de résoudre tout problème construit à l'aide d'une formulation quelconque issue du Générateur de Formulations et d'une géométrie. Des propriétés physiques, celles dont la formulation a besoin, sont ensuite affectées, à l'aide d'un troisième programme, à la géométrie voulue.

Ainsi, contrairement à un programme destiné à résoudre un certain nombre de formulations prédéterminées, notre Solveur est capable de résoudre toutes les formulations, et ceci quelle que soit la géométrie, du moment qu'elles ont été décrites à l'aide du Générateur de formulations. Pour que cela soit possible, nous avons dégagé les grandes lignes d'une résolution numérique d'une EDP (à l'aide de la MIF plus particulièrement), indépendamment de la physique de l'équation. Cette analyse aboutit à une décomposition de la résolution en diverses étapes que nous détaillerons par la suite. Ces différentes étapes sont ensuite formalisées par des classes d'*opérateurs*. Tous les opérateurs de toutes les classes sont décrits formellement dans une Base de Données et sont accessibles pour toutes les opérations utilisant cette base.

Donnons un exemple pour fixer les idées : une des étapes de la résolution est le calcul d'un polynôme d'interpolation en des points appartenant à un élément, il y a donc une classe "Polynômes d'interpolation". Imaginons maintenant qu'un utilisateur veuille rajouter dans la Base de Données une méthode d'interpolation qui n'y était pas : il devra programmer le corps



de son opérateur en utilisant un modèle de fichier vide qui dispose de l'interface standard à respecter pour cette classe. Puis il doit décrire interactivement à l'aide du programme Gestionnaire de la Base de Données le nouvel opérateur en donnant son nom et les attributs demandés (qui sont différents selon chacune des classes d'opérateurs), le programme Gestionnaire se charge ensuite de mettre à jour automatiquement le programme d'appel dynamique des opérateurs pour que le Solveur puisse utiliser le nouveau polynôme qui vient d'être écrit. Ainsi, lors de l'ajout d'un opérateur, l'utilisateur ne fait aucune intervention sur le Solveur, ceci garantissant la fiabilité de ce programme. On peut alors remarquer que, contrairement aux autres techniques utilisables, la croissance de notre système n'est pas fragilisante.

Il est important de faire remarquer que l'ajout d'opérateurs pour décrire une nouvelle formulation n'est pas obligatoire et il est même certain que, au bout d'un certain temps d'utilisation, la Base de Données soit suffisamment riche pour qu'elle contienne tous les opérateurs nécessaire à la majorité des formulations décrites. Ainsi, la Base de Données est aussi une base de connaissances partageables entre les diverses formulations déjà décrites ou à décrire.

En plus de la Base de Données des opérateurs, il y a une Base de Données des variables (ou Dictionnaire des variables physiques) et une Base de Données des modèles de propriétés physiques.

La Base de Données des variables est une liste de noms dans laquelle on puisera les noms des inconnues et propriétés de chaque nouvelle formulation. Cette liste est commune à toutes les méthodes numériques et facilite le couplage entre différentes formulations, même provenant de deux méthodes différentes. En effet, chaque méthode dispose de son Générateur, Solveur généralisé, et dans la version prototype actuelle, de sa propre Base de Données des opérateurs, mais elles partagent le *même* Dictionnaire des variables.

La Base de Données des modèles physiques contient la description des formes ou valeurs que peuvent prendre variables (pour les conditions aux limites) ou propriétés (pour leurs affectations) dans un problème réel. L'intérêt de formaliser et de sauvegarder ces modèles dans un fichier permet d'envisager leur réutilisation pour plusieurs problèmes, ce qui est intéressant notamment à cause de la complexité de certaines modélisations physiques.

Toutes ces Bases de Données vont nous servir à créer et modifier des fichiers de problèmes résolubles par notre Solveur généralisé sur la base de la décomposition fondamentale de tout problème intégral-différentiel :

- Une géométrie
- Une formulation
- Des propriétés physiques

La figure 2.3 de la page suivante schématise les concepts fondamentaux du système en montrant comment interagissent ses divers constituants.

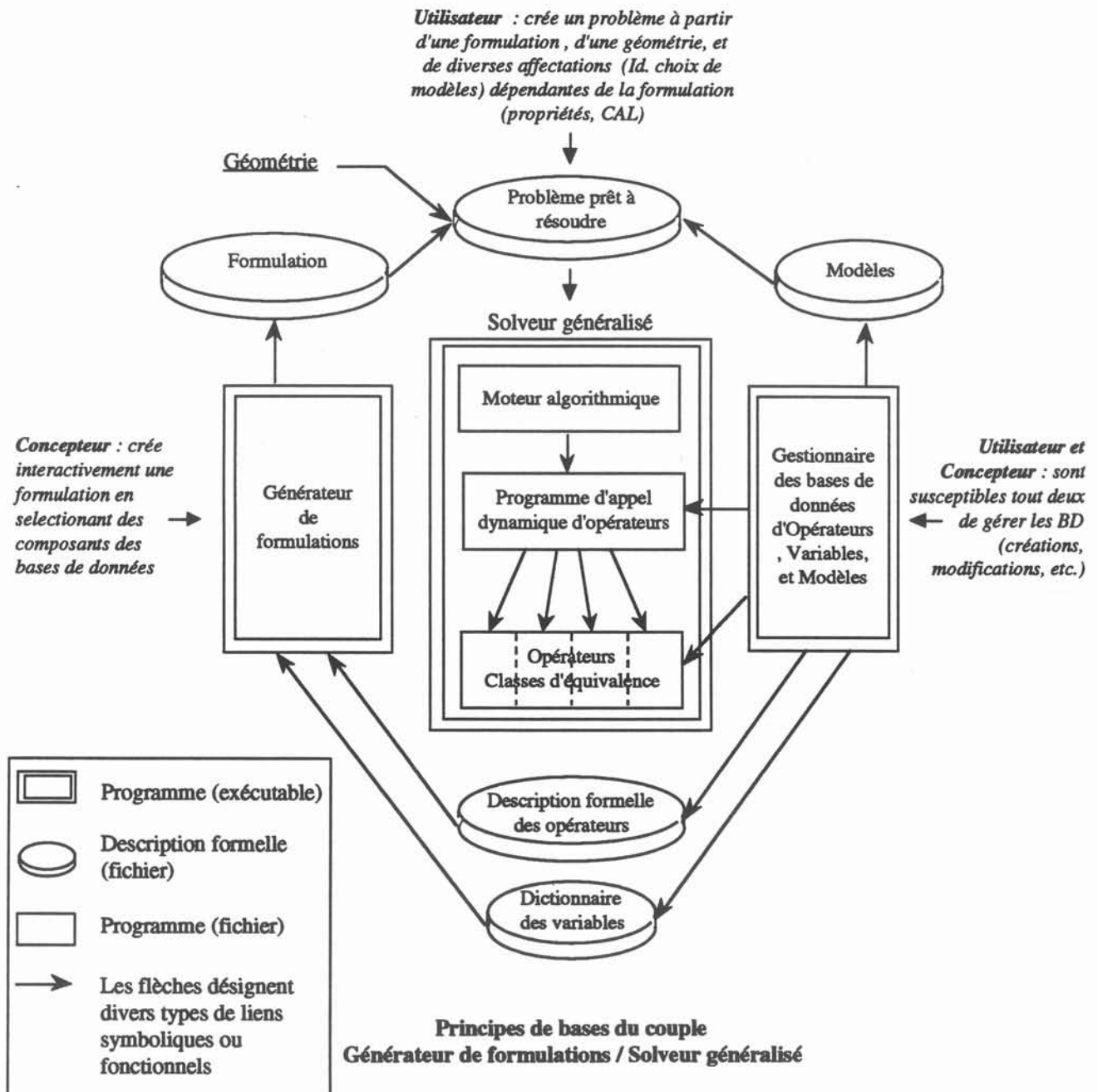


Figure 2.3

On appellera *concepteur* celui qui décrit une nouvelle formulation et on réservera le terme *utilisateur* à celui qui utilise une formulation existante, l'appliquant à des géométries diverses et ne se servant de la Base de Données qu'en édition.

Le concepteur est guidé dans sa tâche par le Générateur par une série de questions lui permettant de décrire sa formulation. Les réponses à ces questions se limitent généralement à un choix d'opérateurs dans la Base de Données. Par exemple, la description de l'inconnue et des propriétés de la formulation consiste tout simplement à choisir les noms des variables que l'on veut utiliser dans le dictionnaire des variables physiques.

La solution que nous venons de décrire représente une étape vers la "solution idéale" dans le classement que nous avons précédemment décrit. En effet, comme nous l'avons déjà remarqué, seul un système expert pourrait apporter la puissance nécessaire au choix optimal de la modélisation physique pour un problème réel. Or, nous sommes convaincus qu'il n'existe pas une formulation mathématique "universelle" comme peuvent le suggérer certains auteurs [32:JIANSHENG], ni même des méthodes numériques valables pour tous les problèmes. C'est pourquoi la solution d'un système ouvert permettant l'introduction de nouvelles formulations et une certaine intervention dans le traitement numérique de ces formulations est une base sur laquelle on pourrait construire un système expert en modélisation physique qui se servirait des diverses décompositions formelles des formulations disponibles, voire qui pourrait en proposer, pour élaborer son expertise. Ainsi, en conservant des parties programmables (les opérateurs) tout en restant dans un cadre formel bien défini, nous avons voulu allier l'adéquation optimale de la programmation à la puissance potentielle d'un système expert pilotant le système. Dans l'immédiat, l'expertise est assumée par celui que nous avons nommé le *concepteur* tandis que *l'utilisateur* dispose d'un produit prêt à l'utilisation pour son problème. Cependant, il est clair que le Générateur pourrait tout à fait être piloté par un système expert en lieu et place du concepteur. Dans l'immédiat, la génération des formulations est grandement facilitée par la possibilité d'utiliser un langage de commande pilotant le Générateur. Les commandes sont des anticipations aux réponses que le concepteur aurait du fournir s'il avait décrit interactivement la formulation. L'existence de cette fonctionnalité permet d'envisager la génération automatique de formulation et représente donc une étape supplémentaire vers le pilotage par un système expert.

La partie délicate de notre travail réside dans le choix et l'élaboration des classes d'opérateurs qui donnent la puissance du système mais aussi ses restrictions puisque qu'elles délimitent les possibilités d'intervention du futur concepteur. Ce choix ne peut se faire qu'à la suite d'une décomposition analytique du déroulement de la résolution d'un problème par la méthode des intégrales de frontières, décomposition que l'on va décrire maintenant.

## 5 Décomposition analytique d'une résolution d'un problème intégrales de frontières

---

### 5.1 Préliminaires

Bien que notre analyse ait été faite dans le cadre de la méthode des intégrales de frontières, les concepts qui en résultent sont très souvent généraux et peuvent s'appliquer à d'autres méthodes numériques, principalement à la méthode des éléments finis. Nous avons constamment gardé à l'esprit la possibilité d'intégration de notre travail dans un cadre plus général axé vers la résolution de problèmes complexes multi-équations, multi-méthodes. C'est pourquoi l'analyse qui suit distingue les caractéristiques générales des particularités propres à la MIF.

Rappelons que le but de la décomposition est de dégager les grandes lignes de la résolution numérique d'un problème afin de définir un cadre formel dans lequel, d'une part un *concepteur* pourra décrire une formulation et, d'autre part un *utilisateur* pourra décrire un problème réel à partir de formulations existantes, de propriétés physiques et de géométries.

La résolution d'un problème (Formulation + Géométrie + Affectations) à l'aide d'une méthode numérique consiste à calculer la valeur des inconnues en un certain nombre de points de la géométrie. Ces valeurs sont issues de la résolution d'un système matriciel carré  $[A]\{X\} = \{B\}$ , de dimension finie grâce à une discrétisation du problème. On explicitera cette notion de discrétisation par la suite.

Le vecteur inconnu  $\{X\}$  peut comporter une ou plusieurs composantes par point. Dans le cas de la MIF, on aura souvent deux composantes puisqu'une formulation MIF fait apparaître la variable d'état et sa dérivée normale à la frontière (Cf chapitre 1).

Les termes de la matrice  $[A]$  comme du vecteur  $\{B\}$  sont issus des intégrales qui interviennent dans la formulation mathématique de l'équation physique. Ces intégrales ne sont pas calculées analytiquement sur leur domaine d'intégration tout entier (on ne connaît pas la formule de leur primitive pour une géométrie quelconque) mais sur des morceaux de ce domaine de formes plus simples, appelés éléments, sur lesquels on sait les calculer soit analytiquement, soit numériquement. La technique qui consiste à ranger les résultats de ces intégrations partielles dans les tableaux globaux du système ( $[A]$  et  $\{B\}$ ) s'appelle l'assemblage.

Les conditions aux limites qui sont, mathématiquement et physiquement, indispensables à la définition complète du problème, permettent, dans une dernière étape, de rendre le système numériquement résoluble.

Cette introduction suffit à dégager les grandes lignes de toute résolution numérique :

- Discrétisation du problème
- Calcul des intégrales du problème discretisé (Intégration)
- Construction du système à résoudre (Assemblage et application des CAL)
- Résolution du système

## 5.2 Discrétisations

Le processus de discrétisation d'un problème recouvre en fait trois discrétisations différentes :

- La discrétisation géométrique
- La discrétisation numérique
- La discrétisation de la formulation

- La discrétisation géométrique : C'est l'une des bases du calcul numérique, toutes méthodes confondues; cela sert à obtenir une représentation discrète du domaine géométrique sur lequel on cherche la solution.

La technique la plus largement utilisée est empruntée à la méthode des éléments finis et consiste à découper le domaine de calcul en éléments. Ces éléments sont des figures géométriques appartenant à des familles topologiques simples (Segment, Triangle, Quadrilatère, Hexaèdre, Tétraèdre, etc.) sur lesquelles sont répartis un certain nombre de points. Ces points servent de base à la discrétisation géométrique du domaine. C'est à dire que la connaissance des coordonnées de ces points et du type topologique des éléments sur lesquels ils s'appuient suffit à déterminer complètement la géométrie. Ainsi, la géométrie de l'élément est décrite par une fonction analytique continue, définie à partir des points qui définissent l'élément.

On appelle les fonctions permettant de définir ainsi la géométrie, les fonctions d'interpolation de la géométrie. La plupart du temps, cette fonction est un polynôme (comme sur l'exemple ci-dessous) d'ordre plus ou moins élevé selon le nombre de points dont on dispose pour la description de la géométrie. On parle alors de polynôme d'interpolation et, pour un élément quelconque décrit par NBP points dans un espace à ND dimensions, on pourra écrire :

$$\forall P \in \text{l'élément}, X_i = \sum_{j=1}^{NBP} \alpha_j(P) x_i^j$$

ou  $X_i$  est la  $i^{\text{ème}}$  coordonnée de P ( $1 \leq i \leq ND$ )

$\alpha_j$  est le polynôme associé au  $j^{\text{ème}}$  point de l'élément. Le polynôme d'interpolation global de l'élément étant défini à l'aide de la somme ci-dessus.

$x_i^j$  est la  $i^{\text{ème}}$  coordonnée du  $j^{\text{ème}}$  point.



- La discrétisation numérique : Supposons que le domaine dans lequel on veut calculer une inconnue soit géométriquement décrit à l'aide de divers éléments, tous identifiables par une collection de points et la fonction d'interpolation rattachée à chaque élément. Ne perdons pas de vue que le but d'une résolution numérique est la connaissance de cette inconnue à l'aide d'un nombre fini de coefficients, généralement calculé à l'aide d'une répartition de points. Comme ces derniers points correspondent bien souvent à des endroits où l'on va calculer la valeur de l'inconnue, il serait logique de penser cette répartition de points (la discrétisation numérique) coïncide avec la discrétisation géométrique. Malheureusement, s'il existe des problèmes pour lesquels cette égalité entre les deux discrétisations est possible, il en existe aussi beaucoup pour lesquels elle est gênante, voire prohibitive.

Les raisons qui poussent à choisir des points servant de support à la discrétisation numérique (on appellera dorénavant ces points des nœuds) différents de ceux servant de support à la discrétisation géométrique (on continuera à appeler ceux-là des points) peuvent être assez diverses. Sans vouloir être exhaustif, citons par exemple la nécessité d'avoir un ordre d'interpolation d'une composante différent d'une autre composante de l'inconnue, ou bien celle de ne pas avoir de nœuds appartenant à plusieurs éléments à la fois (on parlera alors d'éléments numériques discontinus), ou bien encore une technique numérique mettant en œuvre un principe de doublement de nœuds (on aura ainsi deux nœuds distincts pour un point), etc.

On voit donc que, pour un même problème, il y a coexistence de deux discrétisations, a priori distinctes, mais qui font formellement référence à des concepts identiques : les éléments. On distinguera donc les éléments géométriques, issus de la discrétisation géométrique et les éléments numériques, issus de la discrétisation numérique. Nous y reviendrons lors de la description du Générateur.

- La discrétisation de la formulation : Comme on l'a vu dans le chapitre 1, quand on passe de l'équation physique à la formulation mathématique, on est amené à choisir une famille de fonctions sur lesquelles on projette l'équation à résoudre. Ces fonctions (tests, de pondération, ou bien encore fonctions de projection) sont choisies de telle manière que l'équation de départ, valable en tous les points du domaine, ne soit écrite qu'en un nombre fini de points : ce nombre fixe le nombre d'équations du système à résoudre, ou plus pratiquement le nombre de lignes de la matrice [A]. On appellera cette discrétisation, la discrétisation de la formulation et on réservera dans la mesure du possible le terme équation à la notion numérique, issue de la discrétisation de la formulation. Par exemple, dans une méthode des éléments finis discrétisée par une technique de Bubnov-Galerkine, qui est celle la plus souvent utilisée dans Flux-Expert, les fonctions de projection sont les fonctions d'interpolation numérique, c'est à dire que la formulation est calculée aux nœuds du domaine, les équations sont donc confondues avec les nœuds. Par contre, dans la méthode des intégrales de frontières, les fonctions de projection



sont des fonctions de Green choisies de telle façon que la formulation intégrale soit calculée aux pôles de ces fonctions. C'est donc le choix des pôles des fonctions de Green qui détermine les endroits où l'on écrit la formulation, c'est à dire les équations.

### 5.3 Intégrations

Toute formulation faible aboutit à des calculs d'intégrales. Nous distinguerons deux grands types d'intégrales : les intégrales qui font intervenir l'inconnue d'une manière directe ou par son gradient que nous appellerons *intégrales matricielles* car elle fournissent des termes à la matrice [A] et celles qui ne font pas intervenir l'inconnue que nous appellerons *intégrales vectorielles* car elles fournissent des terme au vecteur [B]. Ces intégrales sont calculées sur les éléments qui discrétisent la géométrie. La fonction que l'on doit intégrer est a priori quelconque, appelons la  $f$ , elle peut dépendre de la géométrie, d'autres propriétés physiques, ou même de l'inconnue d'une manière indirecte. Dans ce dernier cas, l'équation est non-linéaire. En plus de la fonction  $f$  que l'on doit naturellement intégrer, toutes les fonctions issues des trois discrétisations décrites ci-dessus interviennent. Ainsi, si  $\{ \psi_i \}$  est la famille de fonctions de projection et  $\{ \alpha_j \}$  la famille des fonctions d'interpolations numériques, et  $\{ \beta_k \}$  la famille des fonctions d'interpolations géométriques, on pourra écrire :

$$a_{ij} = \int_{\text{k}^{\text{ième}} \text{élément}} \{ \psi_i \} f \{ \alpha_j \} d(\beta_k)$$

pour un terme matriciel  $[i,j]$  qui est la contribution du  $j^{\text{ème}}$  nœud, appartenant au  $k^{\text{ième}}$  élément, à une intégrale matricielle, vue de la  $i^{\text{ème}}$  équation. Les fonction d'interpolations géométriques  $\{ \beta_k \}$  interviennent dans le calcul de l'élément différentiel  $d(\beta_k)$  et peuvent également intervenir dans la fonction à intégrer par exemple lorsque des dérivées doivent être prises en compte ou bien pour calculer une dimension caractéristique de l'élément.

De même on aura :

$$b_i = \int_{\text{k}^{\text{ième}} \text{élément}} \{ \psi_i \} f d(\beta_k)$$

pour un terme vectoriel. Bien que l'indice  $j$  n'apparaisse pas, il n'est pas exclu pour autant que les fonctions d'interpolation numérique interviennent si la fonction  $f$  dépend de l'inconnue.

Si toutes les fonctions qui interviennent dans ces intégrales ont des comportements réguliers, il est possible d'envisager une méthode d'intégration numérique très générale. Par contre, quand l'intégrale comporte des singularités, il faut utiliser une technique adaptée au

traitement de cette singularité. Or, la MIF introduit des fonctions de Green qui sont intrinsèquement singulières en leur pôle de définition. Comme, généralement, les pôles appartiennent à la discrétisation du problème, il y a des intégrales singulières. On reviendra plus en détail sur les solutions que nous avons apportées sur le traitement général des singularités et des discontinuités au paragraphe suivant, mais nous voyons d'ores et déjà qu'il est nécessaire de prévoir des *traitements différenciés d'intégrales*. Le traitement le plus fréquent est le traitement numérique (celui qui est utilisé presque toujours dans Flux-Expert) basé une technique de points d'intégration mais il parfois indispensable de faire appel à un traitement analytique ou même un traitement mixte numérique-analytique.

#### 5.4 Construction du système

Une fois que les intégrales sont calculées, il faut les assembler pour construire le système à résoudre. Si la forme du système que l'on doit obtenir est a priori  $[A]\{X\} = \{B\}$  où  $[A]$  est une matrice carrée (car il doit y avoir avant résolution autant d'équations que d'inconnues !), il est beaucoup plus confortable de passer par une étape intermédiaire plus proche de la formulation de départ, sur laquelle on n'a pas encore appliqué les conditions aux limites. Ainsi, dans l'exemple 2.3 du §1, la forme matricielle de la formulation est :

$$[H] \{T\} + [G] \left\{ \frac{\partial T}{\partial n} \right\} = 0 \quad (2.1)$$

et ce n'est qu'après application des conditions aux limites sur  $T$  ou  $\frac{\partial T}{\partial n}$  que le système se réduira en sa forme résoluble ( $[A]\{X\} = \{B\}$ ).

La construction d'un système comprendra donc deux étapes :

La première étape consiste, à partir d'un certain nombre de matrices et de vecteurs, à construire un système matriciel que l'on appelle système matriciel canonique, comme le système (2.1). Nous appellerons les matrices et les vecteurs servant à la construction de ce système canonique *les tableaux constitutifs* (respectivement les matrices constitutives et vecteurs constitutifs). Les tableaux issus du système canonique sont les *tableaux canoniques*. Les matrices constitutives comme les matrices canoniques sont multipliées par des composantes de l'inconnue globale regroupées dans des *vecteurs inconnus*.

La deuxième étape consiste à appliquer les conditions aux limites au système canonique pour obtenir le système  $[A]\{X\} = \{B\}$ .

### 5.5 Résolution du système

Nous ne nous sommes pas penchés en particulier sur ce problème, bien qu'il soit important. Il existe une recherche abondante portant sur la résolution des systèmes d'équations algébriques et des algorithmes éprouvés sont disponibles dans des bibliothèques numériques spécialisées.

## 6 Le Traitement des singularités

La décomposition établie ci-dessus est suffisamment générale pour être valable, à peu de chose près, quelle que soit la méthode numérique employée. Par contre, en ce qui concerne le traitement des singularités, qu'elles soient intrinsèques à la méthode ou bien conséquences de la physique (Cf §1.5.4), les solutions numériques proposées sont particulières à la MIF. Rappelons les trois règles que l'on avait déduites de l'analyse de la prise en compte des singularités en §1.5.4 :

- On doit pouvoir créer des points géométriques où une grandeur pourra être définie plusieurs fois (concept du multi-nœud);
- On doit disposer de toute la latitude possible pour choisir les localisations des nœuds (endroits où les inconnues sont calculées) et des équations (endroits où sont situés les pôles des fonctions de Green);
- On doit dynamiser complètement la manière dont les intégrales peuvent être calculées, c'est à dire prévoir qu'une intégrale peut être calculée de plusieurs manières différentes et même faciliter la création de nouvelles techniques de calcul.

La première règle n'a qu'une incidence sur les structures de données du logiciel qui doivent être prévues pour supporter ce concept du multi-nœuds.

La troisième règle implique l'introduction d'une notion de traitement du calcul des intégrales, dont nous avons déjà souligné la nécessité en 2.2. L'introduction concrète de la notion de traitement sous forme d'opérateur permet de répondre au besoin d'évolutivité et d'adaptabilité du code face à la diversité des solutions possibles (Cf §1.5.4), dans la mesure où aucune de ces solutions n'a la prétention de résoudre tous les cas possibles. Il n'y a donc pas de traitement général des singularités envisageable dans l'optique que nous nous sommes fixée.

La deuxième règle est a priori plus délicate à conceptualiser. En ce qui concerne les nœuds, une localisation "réglable" implique d'une part l'ajout possible de tous types d'éléments dans le système (sur lesquels reposent la répartition des nœuds) et d'autre part des structures de

données internes indépendantes du type d'élément. Par contre, en ce qui concerne les équations, on ne peut plus se reposer sur ces seuls éléments puisqu'une équation est liée à un pôle de la fonction de Green, qui est une notion ponctuelle et a priori indépendante de la localisation des nœuds. On peut cependant remarquer que seules les singularités empêchent une totale correspondance nœuds-équations, on pourra donc partir du principe suivant : la répartition préliminaire des équations possibles est confondue avec celle des nœuds.

On peut alors envisager deux étapes pour déterminer les équations définitives. La première étape permet de décider, en un point, quels sont les équations que l'on veut conserver et de quel type elles sont. Ensuite, la deuxième étape consiste, élément par élément, à déplacer les équations qui doivent l'être à l'issue de la première étape.

Toute l'analyse menée au cours de ce chapitre, associée aux principes énoncés dans le chapitre précédent, a permis l'écriture d'un logiciel prototype : Green-Expert. L'objet du chapitre suivant est de présenter ce logiciel dans son fonctionnement actuel.

## CHAPITRE 3 : GREEN-EXPERT, ARCHITECTURE ET FONCTIONNEMENT

---

### 1 Introduction

---

Construction, résolution, et exploitation sont les trois phases du déroulement de la simulation numérique d'un problème donné.

Green est le programme qui gère la construction d'un problème de simulation numérique dans notre système. Ce programme dispose de quatre modules qui sont destinés chacun à une étape précise. Les quatre modules sont :

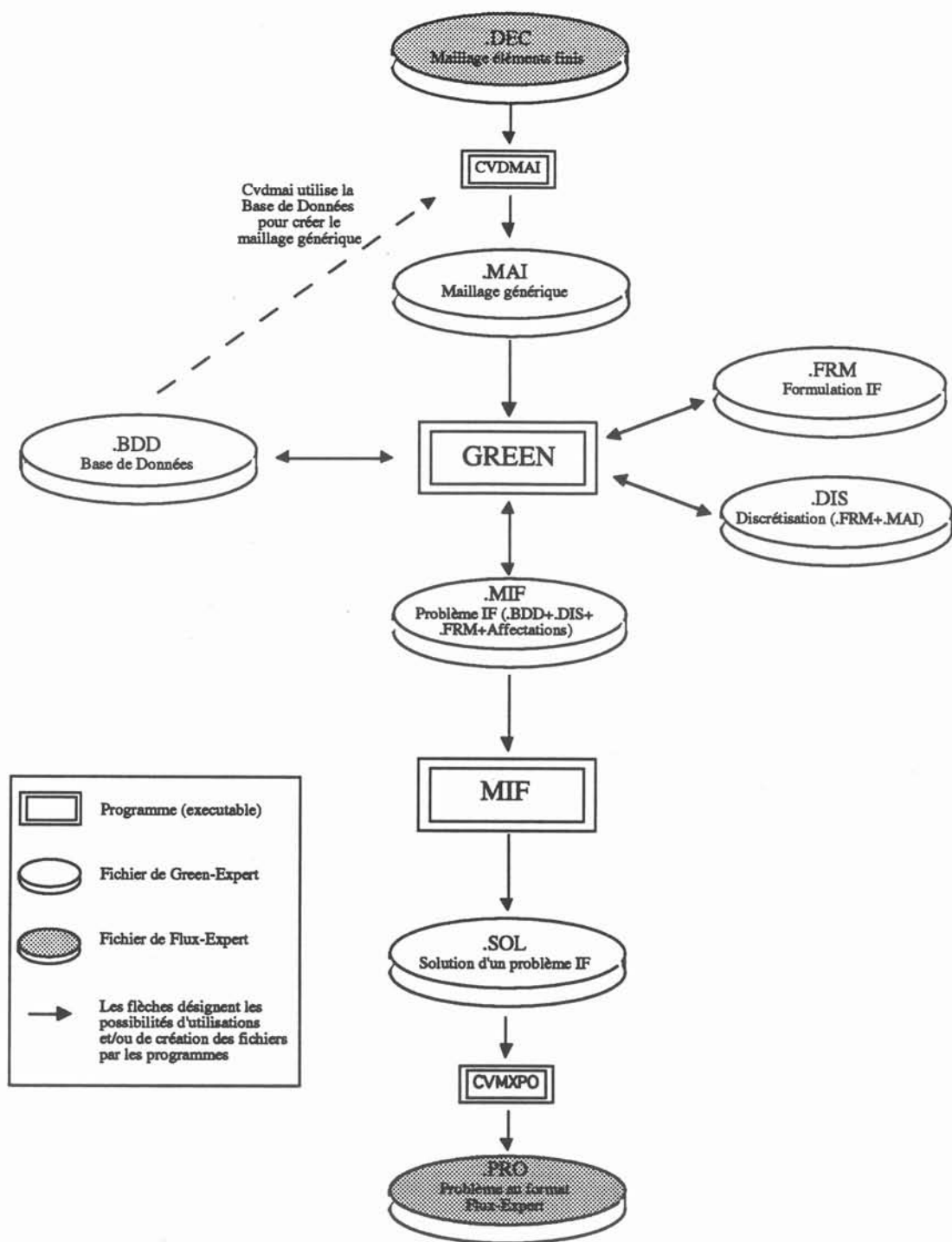
- |                                         |                                      |
|-----------------------------------------|--------------------------------------|
| • La gestion de la Base de Données      | (Fichiers : Bases de données ; .BDD) |
| • La création de nouvelles formulations | (Fichiers : Formulations ; .FRM)     |
| • La description des discrétisations    | (Fichiers : Discrétisations ; .DIS)  |
| • La description des problèmes          | (Fichiers : Problèmes ; .MIF)        |

Rappelons ici que parmi ces quatre étapes, seule la quatrième est véritablement obligatoire puisqu'il se peut que l'on utilise une formulation déjà existante, sans avoir à enrichir la Base de Données, et que l'on dispose d'une discrétisation déjà décrite. La dernière étape consistera donc à décrire les propriétés physiques du problème.

Mif est le programme qui est chargé de résoudre les problèmes issus de Green; c'est le solveur généralisé de Green-Expert. Il ne comporte pas de partie interactive autre que celle qui demande le nom du fichier problème qu'il doit résoudre et le nom du fichier solution qu'il doit sauvegarder. Le fait que ce module soit séparé des tâches de description permet d'envisager l'utilisation du solveur sur une machine plus puissante, éventuellement délocalisée, alors que les modules interactifs ne nécessitant pas beaucoup de puissance peuvent rester sur des machines bas de gamme.

En plus des deux programmes exécutables cités ci-dessus, le système comporte deux utilitaires d'interfaçage amont et aval permettant à Green-Expert de résoudre des problèmes sur les maillages éléments finis utilisés par Flux-Expert (Cvdmai) et de générer des fichiers compatibles avec le Post-Processur éléments finis de Flux-Expert (Cvmxpo). La figure 3.1 schématise l'organisation des fichiers et programmes de Green-Expert.

Les paragraphes suivants décrivent le fonctionnement de Green-Expert module par module ainsi que les interactions entre ces différents modules.



**Organisation des fichiers de Green-Expert**

**Figure 3.1**



## **2 La Base de Données et son gestionnaire**

---

### *2.1 Généralités*

La Base de Données est le cœur du système, tous les autres fichiers y font référence. Elle comporte la description formelle de tous les opérateurs qui appartiennent à une des classes issues de la décomposition analytique décrite au chapitre 2.

La Base de Données comprend deux types de classes d'opérateurs différents :

- Les opérateurs non-programmables (cinq classes)

- Les Identificateurs de Propriétés.
- Les Modèles de Propriétés
- Les Familles Topologiques
- Les Éléments de Référence.
- Les Possibilités de répartition de Points d'Intégration.

La description d'un tel opérateur consiste à répondre à une série de questions, dépendantes de chaque classe, qui permettent de caractériser les attributs du nouvel opérateur. On détaillera un peu les attributs de ces opérateurs par la suite.

- Les opérateurs programmables (quinze classes)

- Les Op. de Description Géométrique.
- Les Op. Polynômes d'Interpolation.
- Les Op. Poids/Points d'Intégration.
- Les Op. Propriétés.
- Les Op. Arguments de Noyau.
- Les Op. Noyaux Singuliers.
- Les Op. Critères de Traitement.
- Les Op. Pilotes de Traitement.
- Les Op. Intégrants.
- Les Op. Coefficients de Collocation.
- Les Op. Singuliers.
- Les Op. de Création d'Équation.
- Les Op. de Déplacement d'Équation.
- Les Op. Construction de Système.
- Les Op. de Résolution de Système.

La description d'un tel opérateur comporte une première phase similaire à celle des opérateurs non-programmables et une deuxième phase qui consiste à écrire le programme correspondant à l'opérateur décrit. L'appel de ce programme par le solveur généralisé est rendu possible par l'écriture automatique du programme d'appel dynamique des opérateurs (Cf §2) par le programme Gestionnaire de la Base de Données. L'interface de tous les opérateurs appartenant à une même classe est identique, cela étant facilité par l'existence d'un opérateur "vide" disposant de cette interface standardisée pour chacune des classes d'opérateurs programmables.

Dans tous les fichiers gérés par Green-Expert, (Cf [fig 3.1](#)), les références à l'un quelconque de ces opérateurs s'effectuent par leur nom. Cela permet d'assurer la portabilité ainsi qu'une certaine pérennité des fichiers puisque, à leur lecture, une correspondance est faite entre les noms des opérateurs contenus dans le fichier lu et les noms de la Base de Données courante. De plus, toute formulation décrite sur un site est ainsi facilement implantable sur un autre site.

Les paragraphes suivants détaillent le rôle et le fonctionnement de tous ces opérateurs. Cependant, pour que la lecture soit plus aisée, on les a regroupé par groupes fonctionnels; il est donc normal de trouver quelques redondances.

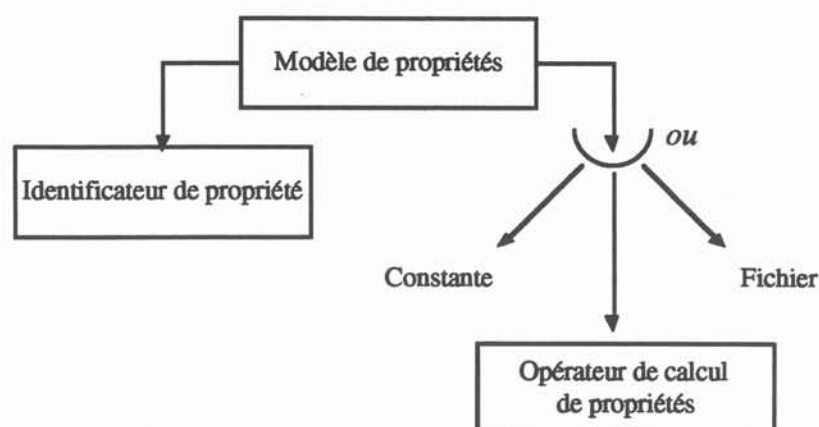
## 2.2 Les propriétés

Les propriétés physiques (dans lesquelles on classe aussi les variables des équations) sont gérées par les classes suivantes :

- La classe des Identificateurs de Propriétés est une liste de noms dans laquelle on devra puiser pour déclarer une inconnue ou une propriété d'une nouvelle formulation. Les noms de cette liste recouvrent des grandeurs qui peuvent être scalaires ou vectorielles.
- La classe des Modèles de Propriétés contient les descriptions formelles de valeurs que peuvent prendre les Identificateurs de Propriétés.
- Les opérateurs de Calcul des Propriétés : Ce sont les opérateurs qui permettent de calculer des propriétés physiques qui ne sont pas constantes

Ainsi, comme le schématise la [figure 3.2](#), une propriété calculée par un opérateur est l'assemblage de ces trois opérateurs. Détaillons cette figure : la classe des Identificateurs de Propriétés correspond à ce que nous avons appelé le Dictionnaire des Variables physiques dans le §2, c'est une classe d'opérateurs non-programmables. Elle a un caractère particulier dans l'optique du couplage entre des systèmes basé sur des méthodes différentes comme le sont Flux-Expert et Green-Expert, ou plus simplement entre deux formulations issues du même

Générateur. En effet, que le couplage se fasse par l'intermédiaire d'une inconnue ou par une propriété, on fera toujours référence à un, ou plusieurs, nom de variable commun aux deux formulations que l'on veut coupler.



Une propriété et ses références possibles  
à la Base de Données

Figure 3.2

Un modèle de propriété est la représentation formelle de chaque valeur d'une propriété, ou de la manière d'obtenir cette valeur. Chaque modèle a un nom et un type, sa description dépend de son type. On dispose actuellement de trois types de calculs possibles :

- Constant
- Fonction calculée par un opérateur de propriétés
- Tabulation stockée dans un fichier.

Parmi ces trois types, seul le type fonction nécessite un choix dans une nouvelle classe d'opérateurs : la classe des opérateurs de propriétés. Une telle classe d'opérateur est indispensable si l'on veut envisager des modélisations complexes. De plus c'est souvent dans ce domaine qu'un utilisateur de code de simulation numérique aura le plus d'interventions à faire. Les opérateurs permettent de calculer des propriétés avec un nombre quelconque de composantes et qui peuvent dépendre de l'inconnue, de la géométrie, du temps, de paramètres constants ou même d'autres propriétés. A titre d'exemple, la figure 3.3 montre la description complète de la chaîne décrite précédemment pour un modèle "fonction", c'est à dire faisant référence à un opérateur de propriété. L'exemple choisi est celui d'un modèle servant à calculer les pertes Joules pour un matériau conducteur dans une formulation magnétique comprenant le potentiel vecteur (représenté à l'aide de l'identificateur AV dans l'exemple).

## Exemple complet de description d'un modèle fonction avec la programmation de l'opérateur associé

## Modèle dans la BD

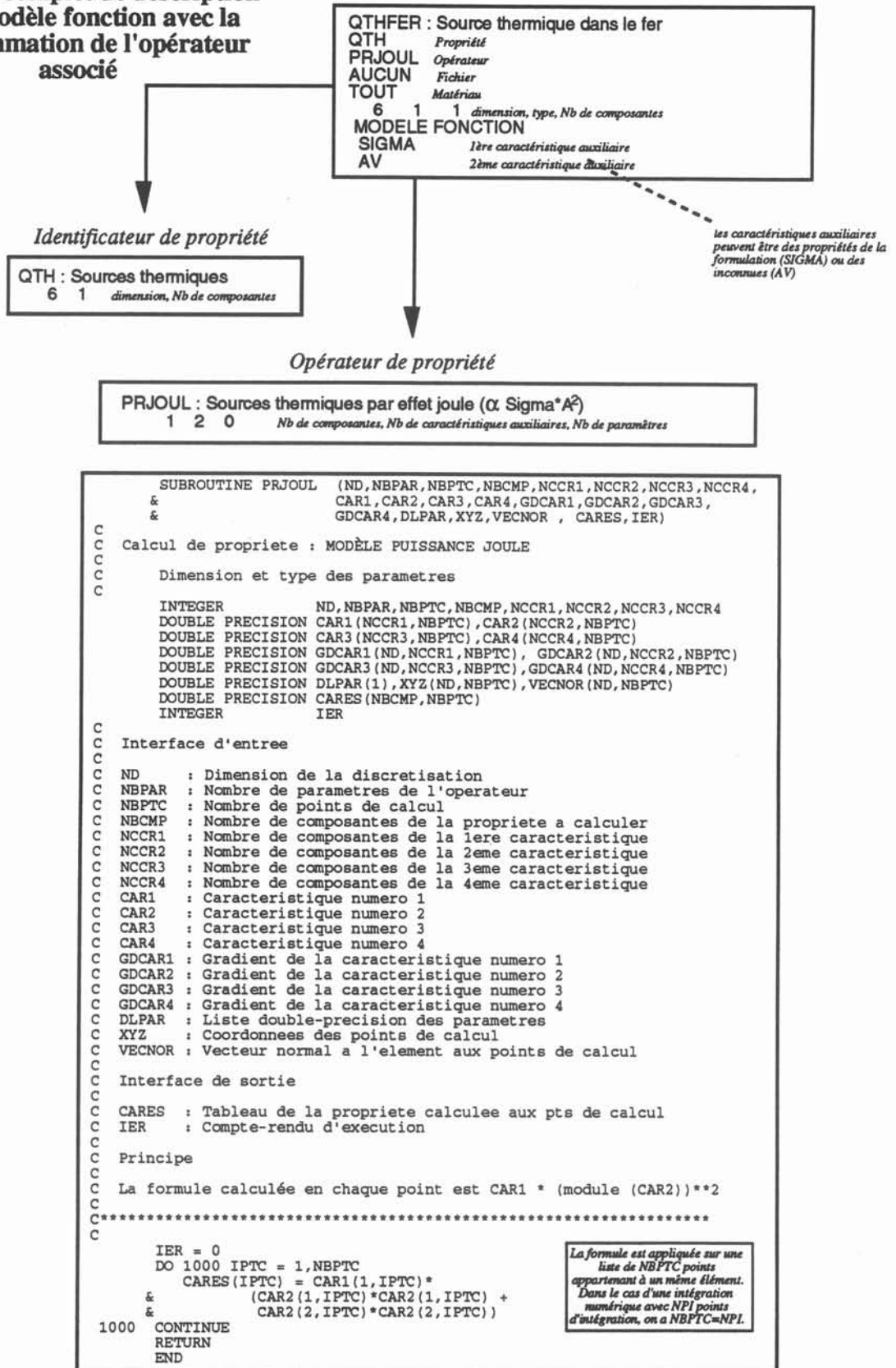


Figure 3.3

### 2.3 Les Éléments de Référence

Un Élément de Référence est une forme géométrique sur laquelle sont ordonnés et positionnés des points dans un repère local. Ces points servent de support à diverses fonctions d'interpolation qui seront la plupart du temps des polynômes.

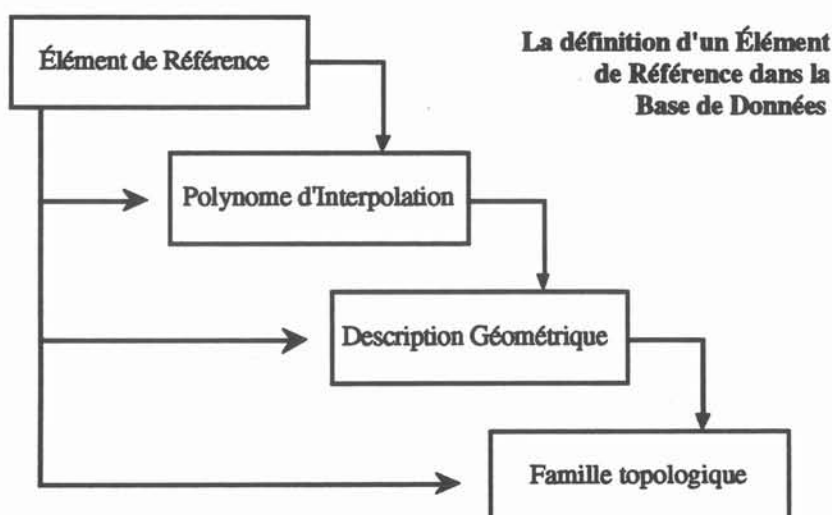
Les notions qui participent à la définition d'un élément de référence sont elles-mêmes décrites dans la Base de Données, ce sont les notions de Familles Topologiques, de Description Géométrique, et de Polynômes d'Interpolation.

- Les Familles Topologiques : C'est une classe d'opérateurs non-programmables, elle permet de faire référence explicitement à une notion souvent utilisée mais toujours implicite, celle de la forme géométrique d'un élément. On aura ainsi les familles topologiques : Segment, Triangle, Hexaèdre, etc. Cela est déterminant pour certains algorithmes tant au niveau de l'intégration que celui du post-processing.

- Les Opérateurs de Description Géométrique : C'est une classe d'opérateurs programmables, le programme étant chargé de renvoyer une liste ordonnée de coordonnées de points. Ces coordonnées sont données dans repère de référence (la plupart du temps compris entre -1 et +1 pour toutes les dimensions).

- Les Polynômes d'Interpolations : C'est une classe d'opérateurs programmables. Le programme est chargé de renvoyer, pour un certain nombre de points de calcul dans l'espace de référence, les valeurs des polynômes et leurs dérivées définissant l'interpolation sur un élément.

Ainsi, un élément de référence est l'assemblage de trois opérateurs de la Base de Données comme le schématise la figure 3.4.



**Figure 3.4**

La notion d'Élément de Référence, est une notion clé du calcul numérique, nous allons expliciter plus en détail cette notion.

Comme on l'a vu au chapitre précédent, on peut exprimer les coordonnées globales d'un point quelconque d'un élément en fonction des coordonnées globales de points spécifiques sur lesquels reposent une fonction d'interpolation et des coordonnées locales à l'élément du point recherché. Ce qui est valable pour une information géométrique (les coordonnées) l'est aussi pour une grandeur physique. Ainsi, les fonctions d'interpolation (la plupart du temps des polynômes) permettent d'exprimer toute grandeur (géométrique ou physique) n'importe où dans un élément en fonction de cette même grandeur, mais localisée en des points précis et connus de cet élément. Cette technique revêt une double utilité.

D'une part, il est clair que toute modélisation numérique doit mener à la résolution d'un système comportant un nombre fini d'équations. Cela implique donc que l'on suppose parvenir à une évaluation satisfaisante de l'inconnue à l'aide de la connaissance d'un certain nombre de valeurs issues de la résolution. Dans la plupart des méthodes de résolution numérique, ces valeurs sont celles de l'inconnue recherchée en des nœuds appartenant au domaine d'étude. Ces nœuds forment un ensemble discret sur lequel on va chercher la solution. Cependant, la simple collection de points n'est pas une structure de données suffisamment souple pour pouvoir manipuler des géométries différentes. C'est là, la première utilité de la notion d'élément : c'est la structuration d'une collection de points qui, associée à une fonction d'interpolation, forme la brique de base sur laquelle reposent toutes les discrétisations. Cette structure est manipulable par des algorithmes généraux qui n'ont pas besoin de connaître la globalité de la discrétisation pour fonctionner et sont donc réutilisables pour tous les problèmes.

La deuxième utilité de la notion d'élément se situe au niveau du calcul. Comme on l'a vu au chapitre 2, la modélisation numérique mène à des calculs d'intégrales sur des domaines qui, comme on vient de le voir, sont discrétisés à l'aide d'éléments. Cela mène donc à calculer des intégrales sur des éléments. Comme il serait préférable de ne pas réécrire les programmes d'intégration pour chaque géométrie, les traitements numériques d'intégration sont écrits une fois pour toutes pour des Éléments de Référence et pour obtenir le résultat recherché sur l'Élément Réel, on procède à un changement de variable.

La notion d'Élément de Référence est donc bien une notion clé de notre système puisqu'elle sert aux deux niveaux de la Discrétisation et de l'Intégration. Il est donc primordial que cette notion soit dynamique et donc présente dans la Base de Données pour pouvoir envisager une bonne adaptabilité. La décomposition en opérateurs que nous en avons faite est suffisamment souple pour que l'on puisse imaginer toutes sortes de discrétisations numériques et géométriques.

La [figure 3.5](#) montre un exemple complet d'un élément de référence décrit, tel que le schématise la [figure 3.4](#), avec les programmes correspondants à cette description.



## Élément de référence

SEGQUA : Segment [-1;+1] (2 noeuds puis 1 central)  
 SEGMENT Famille topologique associée  
 POLLN3 Opérateur polynome d'interpolation associé  
 RFSGQU Opérateur de description géométrique associé  
 1 Type de l'élément (1: régulier, 2: singulier)

## Opérateur polynome d'interpolation

POLLN3 : Lagrange sur Segment quadratique (3 noeuds)  
 RFSGQU Opérateur de description géométrique associé

## Opérateur de description géométrique

RFSGQU : Segment 3 noeuds (ordre 2) [-1;+1]  
 SEGMENT  
 2 3 1 Dimension, Nb de points en totalité, Nb de points par localisation

```

SUBROUTINE POLLN3 (NBPTC,NDC,NBPD,UVW, ALF,DALF,IER)
C
C polynomes et derivees element lineique a 3 noeuds
C
C Dimension et type des parametres
C
C INTEGER NBPTC,NDC,NBPD
C DOUBLE PRECISION UVW(NDC,NBPTC)
C DOUBLE PRECISION ALF(NBPD,NBPTC), DALF(NDC,NBPD,NBPTC)
C INTEGER IER
C
C interface d'entree
C
C NBPTC : nombre de points de calcul
C NDC : Dimensions des coordonnees
C NBPD : Nbre de points decrivant l'element de reference
C UVW : Coordonnees reduites des points de calcul
C
C interface de sortie
C
C ALF : valeur des polynomes
C DALF : valeur des derivees
C IER : compte rendu d'execution
C
C principe
C
C (1) (3) (2) numero des noeuds
C -1 0 +1 Coordonnees reduites
C
C interpolation par les polynomes de lagrange
C
C *****
C
C DOUBLE PRECISION U
C DOUBLE PRECISION R1, R2, RS2
C
C DATA R1, R2, RS2 / 1., 2., .5 /
C
C boucle sur les points de calcul
C
C DO 1000 K = 1, NBPTC
C U = UVW(1,K)
C
C valeurs des polynomes
C ALF(1,K) = RS2 * U * ( U - R1 )
C ALF(2,K) = RS2 * U * ( U + R1 )
C ALF(3,K) = R1 - U*U
C
C valeurs des derivees
C DALF(1,1,K) = U - RS2
C DALF(1,2,K) = U + RS2
C DALF(1,3,K) = - R2 * U
C
C 1000 CONTINUE
C
C RETURN
C END
  
```

La formule est appliquee sur une  
 liste de NBPTC points  
 appartenant à un même élément.  
 Dans le cas d'une intégration  
 numérique avec NPI points  
 d'intégration, on a NBPTC=NPI.

## Famille topologique

SEGMENT : Comme son nom l'indique  
 2 Dimension

## Description complète d'un Élément de Référence

Figure 3.5

## 2.4 La création et le déplacement des équations

En ce qui concerne ce que l'on a appelé au §3 la discrétisation de la formulation, la notion d'élément de référence est insuffisante à définir tous les choix possibles pour les emplacements des pôles des fonctions de Green. En effet, bien qu'il soit fréquent, dans une formulation Intégrales de Frontières de placer ces pôles aux mêmes endroits que les nœuds, cela n'a rien d'obligatoire et se révèle même prohibitif dans le cas de singularités numériques comme on l'a vu §1.5.4 et §2.6. Nous avons donc défini deux opérateurs dédiés à la création de la discrétisation de la formulation, c'est à dire la génération des équations du système numérique :

L'opérateur de Création des Équations dispose en entrée de toutes les équations possibles en un point géométrique donné ainsi que les Conditions aux Limites associées et consiste à retourner le type de chacune de ces équations, c'est à dire si elles sont à conserver, à supprimer, à déplacer, etc. Classiquement, pour une formulation IF à l'ordre zéro où il n'y a aucune singularité numérique, on conservera, pour un point géométrique, une équation sur les deux possibles.

L'opérateur de Déplacement des Équations dispose en entrée de toutes les équations, typées par l'opérateur de Création, appartenant à un élément. On est alors capable de déterminer s'il y a des équations à déplacer et s'il y en a, de donner leurs nouvelles coordonnées locales dans l'élément dans lequel on les déplace.

On rappelle que, par abus de langage, on a appelé "équation" la notion qui désigne un pôle de la fonction de Green associé à une ligne particulière des tableaux dans lesquels on va assembler les intégrales, c'est à dire effectivement une des équations d'un système numérique. La définition des équations d'un problème se passe en deux phases qui sont complètement dynamiques via respectivement, les classes d'opérateurs de Création et de Déplacement des Équations, qui sont des opérateurs programmables. Les figures 3.6 et 3.7 montrent un exemple d'affectations de ces deux opérateurs dans un cas simple.

Les opérateurs dépendent en fait de la formulation, ce n'est donc pas à *l'utilisateur* de choisir ces affectations puisqu'elles ont été faites au moment de la conception de la formulation par le *concepteur* de cette formulation.

Considérons donc par exemple un problème à résoudre sur un carré dont la frontière est discrétisée par huit éléments. Les Conditions aux limites sont données sur la figure 3.6. Avec de telles Conditions aux limites, il reste donc six inconnues à déterminer, comme le montre la figure 3.6.

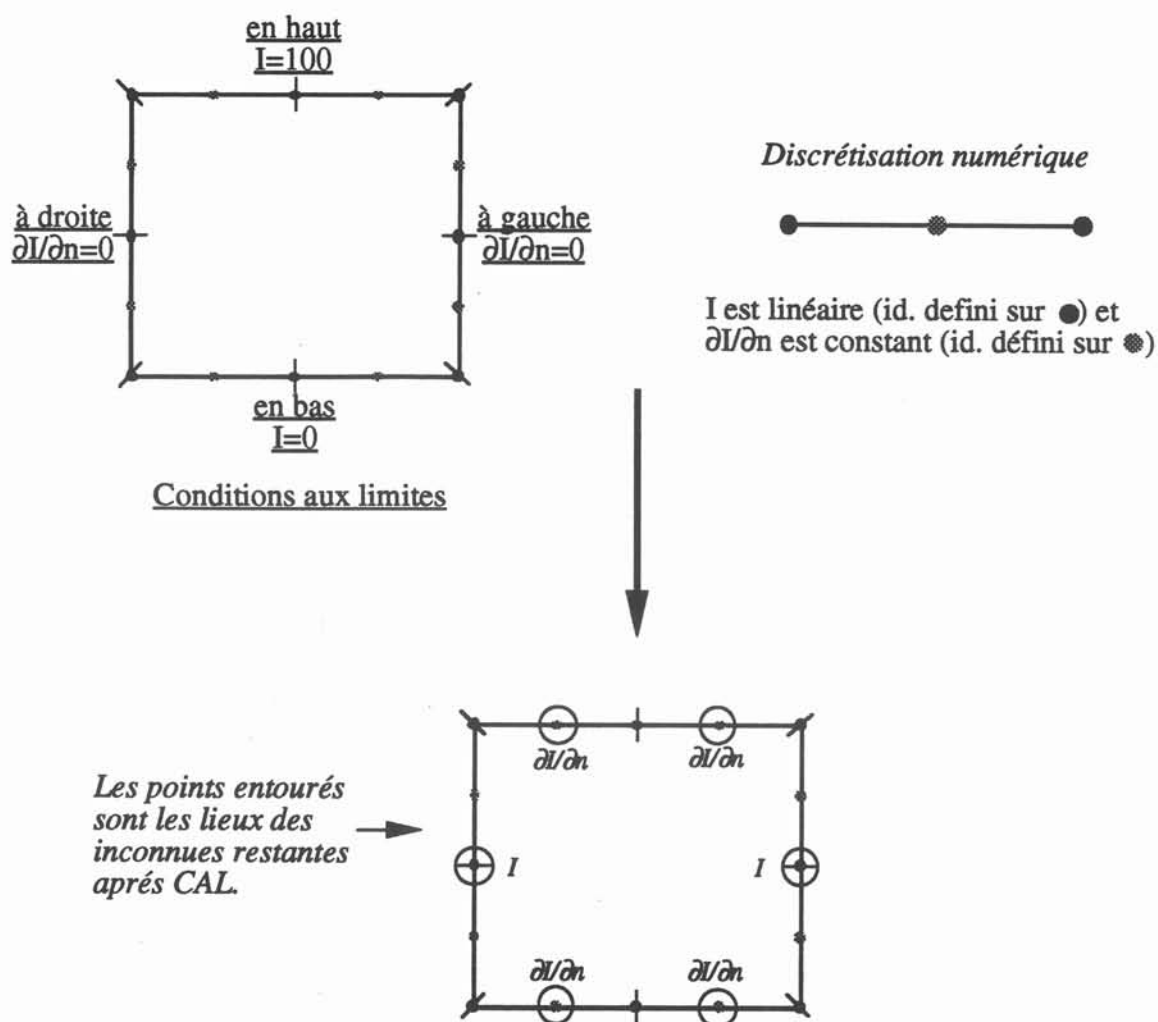


Figure 3.6

La répartition préliminaire des équations est donnée par la discrétisation numérique des deux composantes de l'inconnue :  $I$  et  $\frac{\partial I}{\partial n}$ . Le rôle de l'opérateur de création des équations est de déterminer, parmi toutes ces équations potentielles, quelles sont celles qui vont être effectivement employées. L'opérateur est appelé pour tous les points géométriques, chacun de ces points pouvant correspondre à plusieurs degré de liberté, chaque degré de liberté pouvant correspondre à plusieurs inconnues. Dans notre exemple, il n'y a pas de singularités, chaque point correspond donc à un seul degré de liberté. De plus, comme la discrétisation numérique de  $I$  est disjointe de celle de  $\frac{\partial I}{\partial n}$ , chaque degrés de liberté ne correspond qu'à une inconnue. Ainsi, la règle de création des équations dans ce cas est très simple : il suffit de créer une équation si et seulement s'il n'y a pas de Condition aux limites sur l'inconnue correspondante. On aura ainsi créé autant d'équations qu'il y a d'inconnues, c'est à dire six.

Pratiquement, comme le montre la figure 3.7, la programmation de l'opérateur consiste à remplir avec des codes adéquats (1 pour équation créée, 2 pour équation supprimée) la table IAEQU qui contient toutes les caractéristiques des équations possibles. Comme on peut le voir

dans l'exemple, cette table dispose de toutes les informations utiles à la prise de décision (composante de l'inconnue, numéro du degré de liberté, etc...).

### Opérateur de création des équations

```

EQCOMP : Opérateur classique sans singularités

SUBROUTINE EQCOMP (ND,NBSDOM,NBCOMP,NVCREF,NVCSNG,NUMPT,COOPT,
& IACMP,IASDO,NBDGP,LSDBG,NBEQPO,LEQPO,IRPEQ,
& ICLIM,IAEQU,IER)
C
C   Opérateur de creation des equations sans singularites
C
C   Dimension et type des parametres
C
C   INTEGER          ND,NBSDOM,NBCOMP,NVCREF,NVCSNG,NUMPT
C   DOUBLE PRECISION COOPT(ND)
C   INTEGER          IACMP(5,1),IASDO(12,1)
C   INTEGER          NBDGP,LSDBG(1),NBEQPO,LEQPO(1)
C   INTEGER          IRPEQ(NBDGP,NBCOMP,1),ICLIM(NBCOMP,1)
C   INTEGER          IAEQU(14,1)
C   INTEGER          IER
C
C   Interface d'entree
C   ND      : Dimension de la discretisation
C   NBSDOM  : Nombre de sous-domaine
C   NBCOMP  : Nombre de composante
C   NVCREF  : Numero du vecteur de reference
C   NVCSNG  : Numero du vecteur annexe des relations singulieres
C   NUMPT   : Numero du point de la discretisation
C   COOPT   : Coordonnees du point
C   IACMP   : Attributs entiers des composantes de l'inconnue
C   IASDO   : Attributs entiers des sous-domaines
C   NBDGP   : Nombre de degres de liberte possibles
C   LSDBG   : Liste des degres de liberte possibles
C   NBEQPO  : Nombre d'equations possibles
C   LEQPO   : Liste des equations possibles
C   IRPEQ   : Tableau de la repartition des equations
C   ICLIM   : Codes des conditions aux limites
C
C   Interface d'entree-sortie
C   IAEQU   : Attributs des equations
C
C   Interface de sortie
C   IER     : Compte-rendu d'erreur
C
C   Principe
C   Il ne doit y avoir qu'un degres de liberte possible par
C   point. On cree une equation pour une composante qui n'a
C   pas de condition aux limites
C
C   VOIR LA DOC POUR L'UTILISATION DE IAEQU.
C
C   05/10/93 : Creation
C   .....
C
C   LOGICAL      CAL
C   IER=0
C   Boucle les equations si il n'y pas de singularités
C   IF (NBDGP.EQ. 1) THEN
C     DO 1000 IEQ = 1,NBEQPO
C       NUEQ = LEQPO(IEQ)
C       NUDG = IAEQU(1,NUEQ)
C       NUCPEQ = IAEQU(6,NUEQ)
C       CAL = (ICLIM(NUCPEQ,NUDG) .NE. 0)
C       IF (.NOT. CAL) THEN
C         IAEQU(10,NUEQ) = 1
C       ELSE
C         IAEQU(10,NUEQ) = 2
C       ENDIF
C     1000 CONTINUE
C   ENDIF
C
C   RETURN
C   END

```

Figure 3.7

Dans cet exemple précis, l'opérateur de déplacement des équations est inutile puisqu'il n'y a pas de singularité. Dans le cas où l'on doit traiter des singularités (c'est à dire la présence de plusieurs degrés de liberté pour un seul point géométrique), on peut désigner les équations à déplacer si l'on traite la singularité par cette technique (Cf §1). Un opérateur de déplacement des équations se charge alors de donner les nouvelles coordonnées locales de l'équation dans l'élément où elle est déplacée.

## 2.5 Les opérateurs de Calculs à logique d'Éléments

La notion d'élément de référence ainsi que ces deux précédents opérateurs suffisent à déterminer les trois discrétisations (Cf chapitre 2) d'un problème. Une fois le problème discrétisé, il faut pouvoir calculer les intégrales sur les éléments. Nous avons décomposé le calcul d'une intégrale en deux phases principales. La première phase est celle des calculs à logique d'éléments. La deuxième phase consiste à utiliser les résultats de la première phase pour calculer l'intégrale voulue.

La plupart des opérateurs déjà vus participent, comme on va le voir, aux calculs à logique d'éléments, mais il en existe d'autres :

- Les opérateurs de calcul de poids/points d'intégration : Ce sont des opérateurs programmables qui renvoient, pour une famille topologique donnée, une certaine répartition de poids/points d'intégration selon le nombre de points voulus (éventuellement dans chaque direction pour une méthode produit).

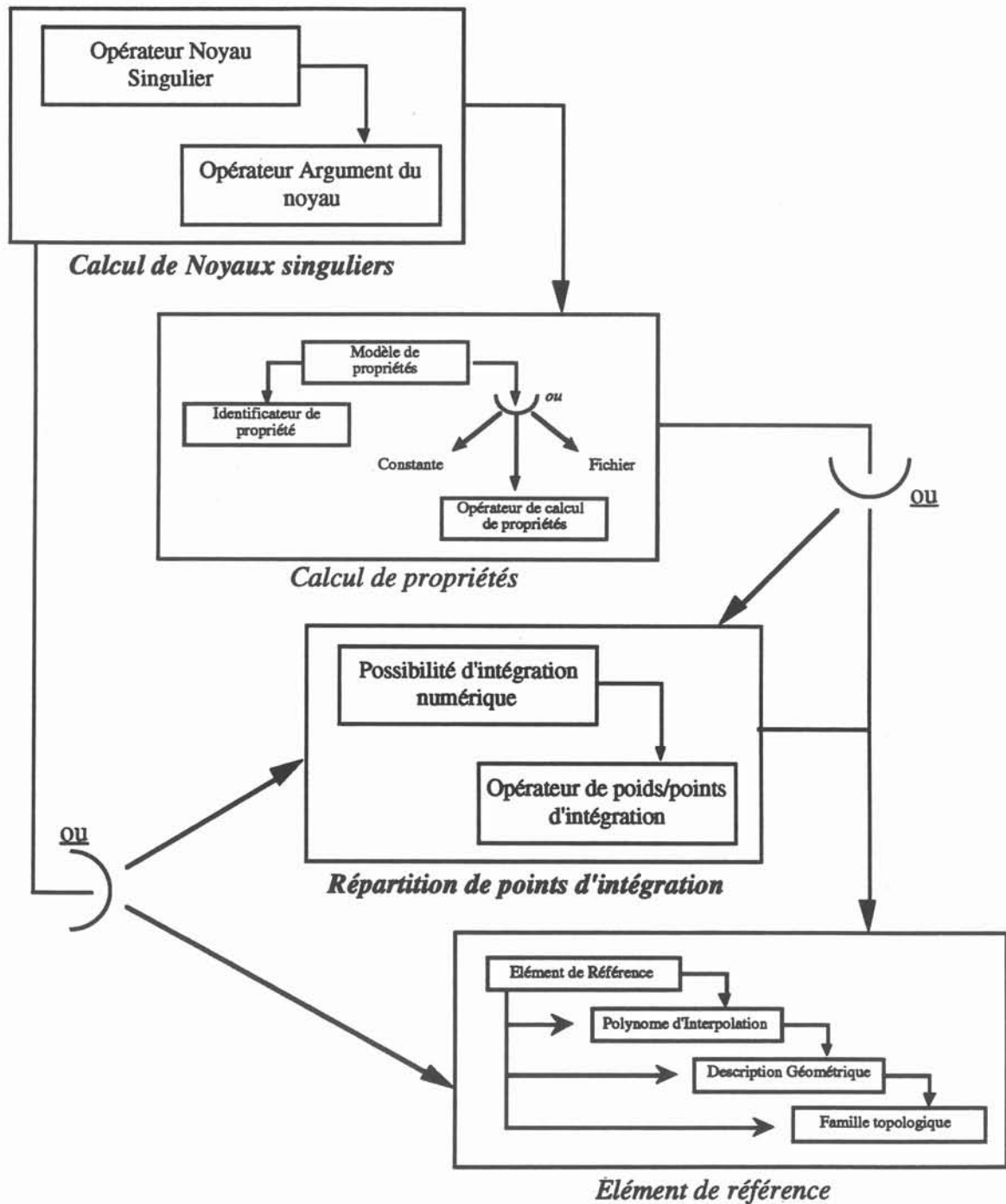
- Les possibilités d'intégration : Ce sont des opérateurs non-programmables. Comme les opérateurs de calcul de poids/d'intégration sont capables de renvoyer plusieurs répartitions possibles, on se sert des possibilités d'intégration pour choisir une de ces répartition. Ainsi, un opérateur de possibilité d'intégration fait référence à un opérateur de calcul de poids/points d'intégration et dispose des attributs nécessaires (Nombre de points dans chaque direction) pour sélectionner sans ambiguïté une des possibilité de l'opérateur de calcul référencé.

- Les opérateurs Arguments de Noyau : Ce sont des opérateurs programmables. Ils renvoient des valeurs qui dépendent non seulement de la répartition élémentaire, mais aussi d'un point extérieur : le pôle d'une fonction de Green. Les valeurs ne sont pas singulières, ce sont par exemple les rayons vecteurs entre le pôle et les points de l'élément :  $\vec{r}_i$

- Les opérateurs Noyaux Singuliers : C'est un opérateur programmable. Son rôle est d'appliquer à l'argument la formule mathématique qui devient singulière quand le pôle est confondu avec un des points de la répartition élémentaire. Typiquement, pour le problème vu en §1.3, le noyau singulier est  $\text{Log}(r_i)$ ,  $r_i$  étant le module du rayon vecteur  $\vec{r}_i$ .

La figure 3.8 schématise ce que nous avons appelé la "boîte à outils" de calculs à logique d'éléments et tous les composants qu'elle implique dans son fonctionnement. Finalement, les opérateurs de calculs à logique d'éléments ont pour tâche de renvoyer des valeurs sur une répartition de points d'un élément. Cette répartition de points peut être intrinsèque à la discrétisation, et donc entièrement définie par un élément de référence, ou bien issue d'un opérateur de possibilité d'intégration. Le calcul des noyaux singuliers peut faire intervenir une propriété, tout comme d'ailleurs les opérateurs de calculs d'intégrales qui peuvent utiliser en

interface d'entrée tout ce que l'on peut obtenir à l'aide de la boîte à outils de calculs à logique d'éléments.



La "boîte à outils" de calculs à logique d'éléments

Figure 3.8

Les figures 3.9 et 3.10 donnent des exemples de description complète d'opérateurs à logique d'éléments.



## Possibilité d'intégration numérique

QUGAU9 : 3 points de Gauss dans chaque direction du carré [-1;+1]				
QUADRILATERE				
Famille topologique				
GAUQ2D				
Opérateur de calcul				
9	3	3	0	Nb total de points, Nb de points dans chaque direction

## Opérateur de calcul de poids/points d'intégration

GAUQ2D : Poids et abscisses de Gauss sur un carré	
QUADRILATERE	
Famille topologique	

```

SUBROUTINE GAUQ2D (ND,NPI,NPIUVW,POIDS,UVW,IER)
C
C  Calcule les poids et abscisses d'integrations sur un carre
C
C  dimension et type des parametres
C
C      INTEGER          ND
C      INTEGER          NPIUVW(2)
C      DOUBLE PRECISION POIDS(NPI) , UVW(2 , NPI)
C      INTEGER          IER
C
C  interface d'entree
C
C  ND      : dimension de l'espace (non utilise ici)
C  NPI     : nombre des points d'integration
C  NPIUVW  : table des Nb.de pts. d'intgr. dans chaque direc.
C
C  interface de sortie
C
C  POIDS   : produit des poids de gauss legendre
C  UVW     : coordonnees des points d'integration
C  IER     : compte rendu d'erreur
C
C  principe
C
C      retourne les poids de gauss et les abscisses des points
C      d'integration sur le carre -1<= u <= +1
C      -1<= v <= +1
C
C *****
C
C      DOUBLE PRECISION PU(10),PV(10),U(10),V(10)
C
C      NPIU = NPIUVW(1)
C      NPIV = NPIUVW(2)
C      CALL GAUS1D(NPIU,PU,U,IER)
C      CALL GAUS1D(NPIV,PV,V,IER)
C
C      DO 1010 I=1,NPIU
C          DO 1000 J=1,NPIV
C              POIDS((J-1)*NPIU+I)=PU(I)*PV(J)
C              UVW(1,(J-1)*NPIU+I)=U(I)
C              UVW(2,(J-1)*NPIU+I)=V(J)
C          1000 CONTINUE
C      1010 CONTINUE
C
C      RETURN
C      END
  
```

## Exemple d'une description d'un couple d'opérateurs Possibilité d'intégration & Calcul de poids/points d'intégration

Figure 3.9

## Opérateur Noyau singulier

**NYINVV : Inversion du module d'un vecteur**  
0 Nombre de caractéristiques nécessaires

```

SUBROUTINE NYINVV (ND,NBPTC,NCARG,ARG,NC1,NC2,
& P1,P2,VECNOR,COOPC , NOYS,IER)
C
C Inverse du module de l'argument vecteur
C
C Dimension et type des parametres
C
C INTEGER ND,NBPTC,NCARG
C DOUBLE PRECISION ARG(NCARG,NBPTC)
C INTEGER NC1,NC2
C DOUBLE PRECISION P1(NC1,NBPTC)
C DOUBLE PRECISION P2(NC2,NBPTC)
C DOUBLE PRECISION VECNOR(ND,NBPTC)
C DOUBLE PRECISION COOPC(ND,NBPTC)
C DOUBLE PRECISION NOYS(NBPTC)
C INTEGER IER
C
C Interface d'entree
C
C ND : Dimension des coordonnees
C NBPTC : Nombre de points de calculs
C NCARG : Nombre de composante de l'argument
C ARG : Argument du noyau
C NC1 : Nombre de composantes de P1
C NC2 : Nombre de composantes de P2
C P1 : Propriete 1
C P2 : Propriete 2
C VECNOR : Vecteur normal a l'element
C COOPC : Coordonnees des points de calcul
C
C Interface de sortie
C
C NOYS : VNoyau singulier aux pts de calcul
C IER : Compte-rendu d'erreur
C
C Principe
C
C L'argument est suppose etre un vecteur.
C On teste donc NCARG=ND avant de calculer le module
C *****
C CHARACTER*6 NOMPRO
C DOUBLE PRECISION RMOD,SHNORV
C NOMPRO='NYINVV'
C IER=0
C RMOD=0.D0
C
C IF (NCARG.NE. ND .OR.
& (ND.NE. 2 .AND. ND.NE. 3)) GOTO 9000
C
C DO 1000 IPTC=1,NBPTC
C Le module du vecteur ...
C RMOD = SHNORV (ND,ARG(1,IPTC))
C On teste tout de meme avant d'inverser
C IF (ABS(RMOD) .LT. 1D-20) GOTO 9010
C NOYS(IPTC) = 1.D0 / RMOD
C 1000 CONTINUE
C
C 999 RETURN
C
C Erreur d'interface
C 9000 IER=41
C CALL ERREUR(IER,1,NOMPRO,' ')
C GOTO 999
C Argument nul
C 9010 IER=31
C CALL ERREUR(IER,2,NOMPRO,' ')
C GOTO 999
C
C END

```

## Opérateur Argument d'un noyau singulier

**ARRY3D : Rayon vecteur en 3D**  
3 0 Nombre de composantes, Nombre de caractéristiques nécessaires

```

SUBROUTINE ARRY3D (ND,NBPTC,NCARG,COONC,COOPC,
& NC1,P1,VECNOR , ARG,IER)
C
C Rayon vecteur (3D) collocation-->pts de calcul
C
C Dimension et type des parametres
C
C INTEGER ND,NBPTC,NCARG
C DOUBLE PRECISION COONC(ND),COOPC(ND,NBPTC)
C INTEGER NC1
C DOUBLE PRECISION P1(NC1,NBPTC),VECNOR(ND,NBPTC)
C DOUBLE PRECISION ARG(NCARG,NBPTC)
C INTEGER IER
C
C Interface d'entree
C
C ND : Dimension des coordonnees
C NBPTC : Nombre de points de calcul
C NCARG : Nombre de composantes de l'argument
C COONC : Coordonnees du noeud de collocation
C COOPC : Coordonnees des points de calcul
C NC1 : Nombre de composantes de la propriete P1
C P1 : Propriete 1
C VECNOR : Vecteur normal a l'element
C
C Interface de sortie
C
C ARG : Argument d'un noyau singulier
C IER : Compte-rendu d'erreur
C
C Principe
C
C Calcul du rayon vecteur 3D entre
C Un noeud de collocation et
C une liste de points de calcul
C appartenant a l'element
C *****
C
C CHARACTER*6 NOMPRO
C NOMPRO='ARRY3D'
C IER=0
C
C On verifie l'interface
C IF (NCARG.NE. 3 .OR. ND.NE. 3) GOTO 9000
C
C DO 1010 IPTC=1,NBPTC
C DO 1000 IAR=1,NCARG
C ARG(IAR,IPTC) = COOPC(IAR,IPTC)-COONC(IAR)
C 1000 CONTINUE
C 1010 CONTINUE
C
C 999 RETURN
C
C Erreur d'interface
C 9000 IER=41
C CALL ERREUR(IER,2,NOMPRO,' ')
C GOTO 999
C
C END

```

## Exemple d'une description d'un couple d'opérateurs Noyau singulier & Argument de Noyaux singuliers

Figure 3.10

La figure 3.9 montre la description et la programmation des opérateurs nécessaires à l'intégration numérique sur un quadrilatère avec 9 points d'intégration (3 dans chaque direction). On remarque que, comme on emploie une méthode dite "produit", l'opérateur de

calcul doit appeler l'opérateur de calcul de dimension inférieure nécessaire, c'est à dire ici les points d'intégration sur un segment.

La figure 3.10, quant à elle, montre la description et la programmation des deux opérateurs nécessaires au calcul du noyau singulier  $\frac{1}{r}$ , où  $r$  est le module du rayon vecteur en 3D.

## 2.6 Les opérateurs de Calculs d'Intégrales

Le terme "intégrale" est ici à prendre dans un sens plus large que "résultat d'une intégration". On a en fait regroupé sous ce terme tout ce qui est destiné à être assemblé dans une des matrices constitutives du système à construire. (Cf §2). Cela regroupe donc les intégrales au sens classique du terme, qu'elles soient matricielles ou vectorielles (c'est à dire, on le rappelle, destinées à être assemblées dans une matrice ou dans un vecteur), les coefficients des relations de continuités entre les sous-domaines (Cf §1), ou les coefficients géométriques de collocation. Cependant, pour ne pas charger inutilement les interfaces des opérateurs et les rendre les plus simples possible à utiliser, nous avons défini trois opérateurs programmables différents pour les trois types d'intégrales citées ci-dessus.

• Les opérateurs Intégrants : Ce sont les opérateurs correspondant à une intégration, au sens classique du terme. Dans la description des ces opérateurs dans la Base de Donnée, figurent les informations suivantes :

- Type de calcul : Analytique ou Numérique
- Nature de l'intégrant : Vectoriel ou Matriciel
- Dimension : 1,2,3 ou sans
- Nombre de Propriétés : 0 à 2

Nous avons vu en §2.2.2 la formule générale des intégrales matricielles et vectorielles, l'interface des programmes correspondant à ces opérateurs reflète cette formule. A titre d'exemple, la figure 3.11 donne la programmation d'un opérateur intégrant numérique. On remarquera que tous les tableaux en entrée sont issus des opérateurs de calculs à logique d'éléments (Noyau, Polynôme, Propriétés, Coordonnées des points de calculs et leur poids d'intégration) bien qu'ils aient pu être affectés par le changement de repère qu'implique le passage de l'élément de référence à l'élément réel.

C'est notamment le cas pour les interpolations numériques et géométriques qui sont affectées par ce changement de repère et permettent d'obtenir le vecteur normal à l'élément aux points de calculs (pour les éléments appartenant à la frontière) ainsi que les poids d'intégration qui sont multipliés par le Jacobien de la transformation au cours d'une procédure classique de l'intégration numérique que l'on peut trouver dans les ouvrages généraux ([4:BREBBIA],[16:DHAT]). Dans notre Solveur, cette étape est complètement prise en charge

par le programme et l'on dispose dans l'opérateur des poids d'intégration déjà multipliés par le Jacobien de la transformation.

```

      SUBROUTINE IMNNOY (ND,NBPTC,NUEL,TNOSG,NBN,ALF,DALF,
&                      VECNOR,POIDS,COOPC,NC1,NC2,P1,DP1,P2,DP2 ,
&                      TBINT,IER)
C
C   Integrant Matriciel Numerique egal au Noyau
C
C   Dimension et type des parametres
C
      INTEGER          ND,NBPTC,NUEL
      DOUBLE PRECISION TNOSG(NBPTC)
      INTEGER          NBN
      DOUBLE PRECISION ALF(NBN,NBPTC),DALF(ND,NBN,NBPTC)
      DOUBLE PRECISION VECNOR(ND,NBPTC)
      DOUBLE PRECISION POIDS(NBPTC)
      DOUBLE PRECISION COOPC(ND,NBPTC)
      INTEGER          NC1,NC2
      DOUBLE PRECISION P1(NC1,NBPTC),DP1(ND,NC1,NBPTC)
      DOUBLE PRECISION P2(NC2,NBPTC),DP2(ND,NC2,NBPTC)
      DOUBLE PRECISION TBINT(1)
      INTEGER          IER
C
C   Interface d'entree
C
C   ND      : Dimension de l'espace
C   NBPTC   : Nombre de points de calcul
C   NUEL    : Numero de l'element
C   TNOSG   : Noyau singulier aux points de calcul ( = Ptc)
C   NBN     : Nombre de noeuds de l'element
C   ALF     : Polynome d'interpolation de la variable (Ptc)
C   DALF    : Derivee du poly. d'interp. de la variable (Ptc)
C   VECNOR  : Vecteur normal a l'element (Ptc)
C   POIDS   : Poids differentiels (Ptc)
C   COOPC   : Coordonnees des points de calcul
C   NC1     : Nombre de composante de la propriete P1
C   NC2     : Nombre de composante de la propriete P2
C   P1      : Propriete 1 aux points de calcul
C   DP1     : Gradient de P1
C   P2      : Propriete 2 aux points de calcul
C   DP2     : Gradient de P2
C
C   Interface de sortie
C
C   TBINT   : Vecteur integre
C   IER     : Compte-rendu d'execution
C
C*****
C
      CHARACTER*6      NOMPRO
      DOUBLE PRECISION SOM
C
      NOMPRO='IMNNOY'
      IER=0
C
C   Boucle sur les noeuds
      DO 1010 IN=1,NBN
        SOM = 0.D0
C
C   Boucle sur les points de calculs
        DO 1000 IPTC=1,NBPTC
          SOM = SOM + ALF(IN,IPTC)*TNOSG(IPTC)*POIDS(IPTC)
1000    CONTINUE
        TBINT(IN) = SOM
1010  CONTINUE
C
      RETURN
      END

```

**Figure 3.11**

Comme on le voit dans l'exemple de la [figure 3.11](#), la formule à programmer ne fait que quelques lignes dans le cas le plus fréquent de l'intégration numérique. En l'occurrence, ici

l'intégrale à calculer est  $\int N \cdot X$ ,  $N$  étant un noyau singulier quelconque et  $X$  une composante de l'inconnue. En pratique, cette intégrale sera calculée sur un élément, pour un pôle du noyau précis. On écrit alors l'interpolation numérique de  $I$  sur cet élément, c'est à dire  $X = \sum_{i=1}^{NBN} \alpha_i x_i$ .

Le calcul de l'intégrale ci-dessus (dite "matricielle" car son résultat sera assemblé dans une matrice), renvoie donc un vecteur composé de  $NBN$  intégrales, chacune d'entre elles correspondant à un nœud de l'élément. La formule de ces intégrales est donc :

$$\int_{\text{éf}} \alpha_i N_j d(\text{éf})$$

- où :
- éf est un élément de frontière
  - $\alpha_i$  est le polynome d'interpolation numérique ( $i$  variant de 1 à  $NBN$ , nombre de nœuds de l'élément)
  - $N_j$  est le noyau singulier calculé aux  $NBPTC$  points de calcul (ici des points d'intégration) de l'élément ( $j$  étant relatif au pôle du noyau)

L'opérateur effectue donc une double boucle : la plus interne, sur  $NBPTC$ , correspond à la somme nécessaire pour l'intégration numérique et l'externe sur  $NBN$  qui correspond au calcul des  $NBN$  intégrales décrites ci-dessus.

• Les opérateurs Singuliers : Ils sont spécifiquement dédiés au calcul des relations singulières (Cf §1.5.4). Leur interface est similaire à celle des opérateurs intégrant analytiques mais, au contraire de ces derniers, la logique du calcul est ponctuelle et non par élément. Elles font généralement intervenir les nœuds qui font référence à un même point géométrique, par exemple pour exprimer une relation de continuité (ou de discontinuité) entre deux facettes de part et d'autre d'une arête ou plus classiquement entre deux sous-domaines de part et d'autre d'une interface comme le montre la figure 3.12.

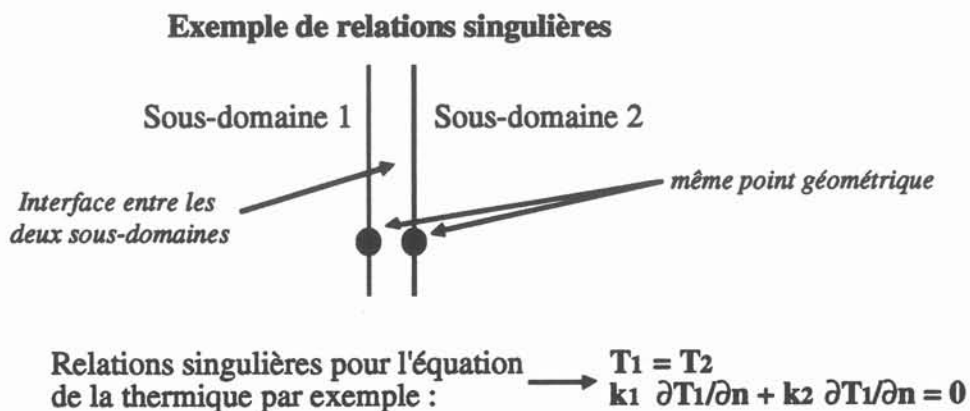


Figure 3.12

• Les opérateurs de coefficient de collocation : Ils sont destinés au calcul de ces coefficients qui apparaissent dans une formulation Intégrales de Frontières quand le nœud définissant une équation (le nœud de collocation) appartient à la frontière. Ils sont le résultat d'un passage à la limite d'intégrales singulières et sont donc des expressions analytiques a priori dépendant de la formulation. Cependant, les fonctions de Green ont des noyaux singuliers dont les comportements, proches de zéro, sont similaires et on constate donc que ce coefficient de collocation dépend directement de, ou même est parfois égal à, l'angle que fait la frontière à l'endroit où on le calcule. Typiquement, pour l'équation de Laplace en 2D, si le nœud de collocation appartient à un angle droit, le coefficient sera égal à  $\frac{\pi}{2}$ , alors qu'en 3D, c'est l'angle solide que l'on devra calculer.

```

      SUBROUTINE CRAPPA (NBTRT,NUMSD,ISECG,LELEQ,NUEL,IAPPAR ,
&
      NUMTRT,IER)
C
C Critere de traitement d'appartenance
C
C Dimension et type des parametres
C
      INTEGER NBTRT,NUMSD,ISECG,LELEQ(1),NUEL,IAPPAR
      INTEGER NUMTRT,IER
C
C Interface d'entree
C
C NBTRT : Nombre de traitements possibles
C NUMSD : Numero du sous-domaine
C ISECG : Numero de la section geometrique
C LELEQ : Liste des elements des equations
C NUEL : Numero de l'element en cours
C IAPPAR : Indicateur d'appartenance de l'equation a l'element
C
C Interface de sortie
C
C NUMTRT : Numero du traitement
C IER : Compte-rendu d'execution
C
C Principe
C
C Le traitement singulier est le numero 1
C Le traitement regulier est le numero 2
C
C*****
C
      CHARACTER*6 NOMPRO
      NOMPRO='CRAPPA'
      IER=0
C
      Dans ce cas tres simple, on se sert de l'indicateur
      d'appartenance et seulement si l'equation est normale
      ou deplacee
      ITYPEQ = IVEQUA(15)
C
      IF (ITYPEQ .EQ. 1 .OR. ITYPEQ .EQ. 3 .OR. ITYPEQ .EQ. 4) THEN
      IF (IAPPAR .EQ. 1) THEN
        NUMTRT=1
      ELSE
        NUMTRT=2
      ENDIF
      ELSE
        NUMTRT=0
      ENDIF
C
      RETURN
      END

```

*On dispose d'une fonction permettant d'accéder aux caractéristiques de l'équation en cours. En l'occurrence, ici, on se sert du type de l'équation.*

**Exemple d'opérateur critère de traitement**

Figure 3.13



## 2.7 Les opérateurs de traitement des intégrales

Comme on l'a déjà fait remarquer, la méthode des Intégrales de Frontières offre une palette très large de possibilités d'intégrales. La notion de traitement d'intégrales est notre réponse à ces différentes possibilités. Le traitement d'une intégrale est accessible à un concepteur de formulations via deux opérateurs complémentaires :

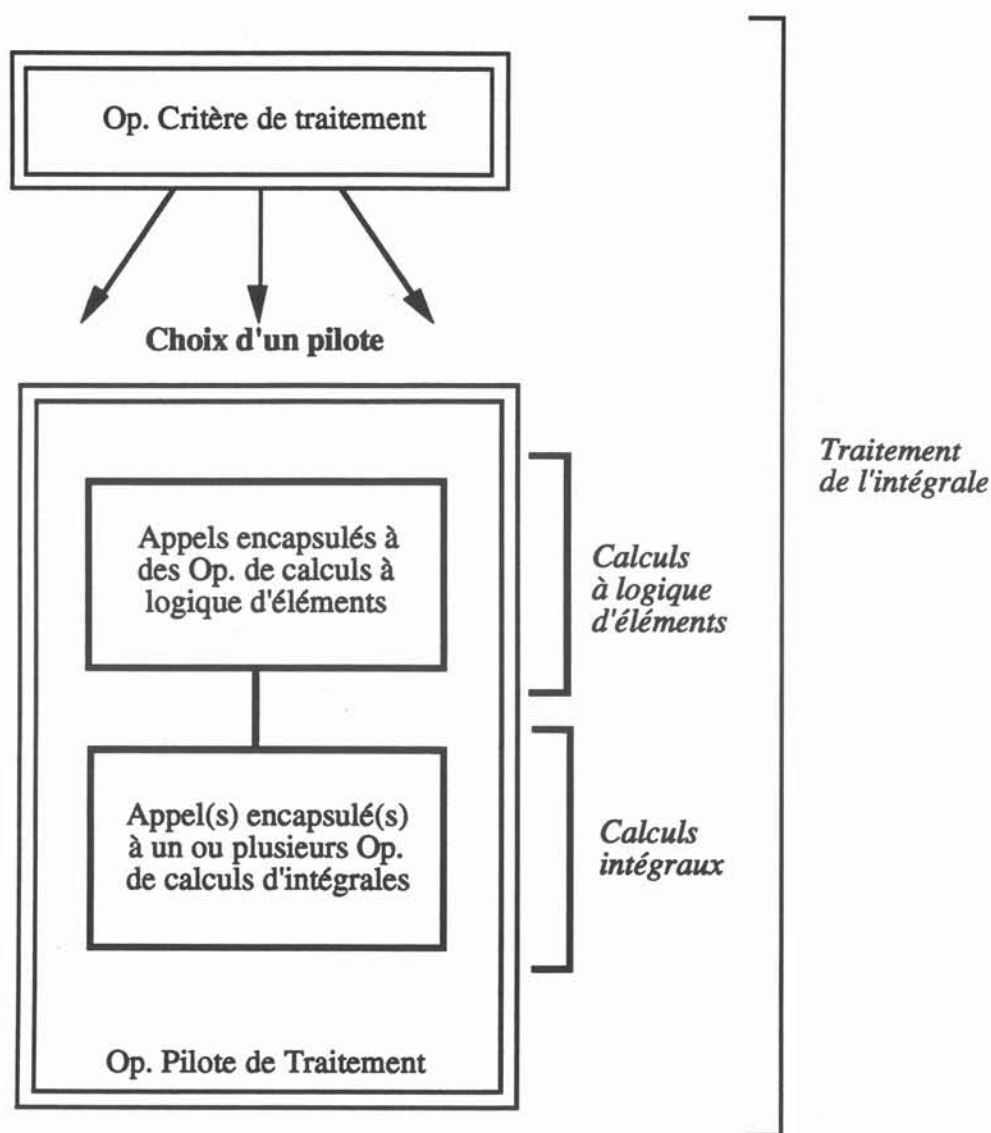
- L'opérateur critère de traitement : C'est un opérateur programmable. Il dispose en entrée de toutes les informations utiles caractéristiques du contexte d'intégration et doit retourner un numéro de traitement suivant ce contexte.
- L'opérateur Pilote de Traitement : C'est aussi une classe d'opérateurs programmables. Les pilotes de traitements sont les programmes chargés de calculer une intégrale avec une technique qui leur est spécifique.

La [figure 3.13](#) donne l'exemple de l'opérateur classique qui permet de choisir un traitement singulier si l'équation en cours appartient à l'élément en cours et le traitement régulier sinon.

La [figure 3.14](#) montre comment s'articule l'utilisation de ces deux opérateurs : Comme on le verra dans la description d'une formulation, une intégrale peut être calculée de plusieurs manières différentes suivant le contexte d'intégration. Nous avons actuellement développé six pilotes de base qui couvrent tous les types d'intégrales mais bien sûr pas tous les traitements possibles. Ces six pilotes sont les quatre pilotes de traitements des intégrales matricielles et vectorielles, chacune d'entre elle pouvant être calculée analytiquement et numériquement, plus les deux pilotes spécifiques de traitement des opérateurs de collocation et singuliers. Le déroulement de tous ces pilotes de traitement a été décomposé en deux sous-phases correspondant aux deux grandes familles d'opérateurs décrites ci-dessus, c'est à dire les opérateurs de calculs élémentaires et les opérateurs de calcul d'intégrales. Il est clair que les pilotes peuvent être délicats à programmer, il touchent des structures de données délicates à manipuler, et il est probable que la majorité des formulations se contentent des six pilotes de bases déjà programmés. Cependant, pour faciliter leur programmation, nous avons encapsulé l'accès à toutes les données nécessaires aux calculs.

Les pilotes de traitement sont les opérateurs qui, comparativement aux autres, interviennent le plus haut dans la chaîne d'intégration.

Une fois l'intégrale calculée, il faut l'assembler soit dans une matrice, soit dans un vecteur.



**Principe de Calcul d'une Intégrale**

Figure 3.14

### *2.8 L'opérateur Construction de Système*

La grande variété des termes obtenus lors de la décomposition matricielle des formulations est matérialisée par cette classe d'opérateurs programmables. La description d'un nouvel opérateur de cette classe consiste à déclarer le nombre de vecteurs inconnus de la formulation, le nombre de matrices, de vecteurs. Le programme correspondant à un opérateur construction de système est chargé de manipulations matrices-vecteurs conduisant à construire les tableaux nécessaires à l'application des conditions aux limites quand il s'agit d'un système destiné à une résolution ou au calcul des inconnues intérieures dans le cas d'un système destiné au post-processing (Cf §3.5). L'intérêt de cette classe d'opérateur est qu'elle permet de décrire

absolument n'importe quelle forme de système en étant indépendant d'une forme générique, comme par exemple  $[H]\{I\} + [G]\{\partial I/\partial n\}$ . Supposons par exemple qu'il soit intéressant de décrire le système matriciel sous la forme :

$$([P]([H_1] + [H_2]))\{I\} + ([G_1] + [G_2])[Q]^{-1}\{\partial I/\partial n\}$$

Alors l'opérateur construction de système recevra en entrée les matrices  $[P]$ ,  $[H_1]$ ,  $[H_2]$ ,  $[G_1]$ ,  $[G_2]$  et  $[Q]$  et sera chargé d'effectuer les opérations matricielles de transposition, d'inversion, de somme et de produit, afin de calculer les matrices :

$$[P]([H_1] + [H_2]) \text{ et } ([G_1] + [G_2])[Q]^{-1}$$

qui permettent de nous ramener à la forme générique traitée par application des conditions aux limites.

Seul deux opérateurs de construction de systèmes sont actuellement programmés :

- Un système de résolution de la forme  $[H]\{I\} + [G]\{\partial I/\partial n\} = \{K\}$
- Un système de post-processing de la forme  $\{C\}\{I_{pp}\} = [H]\{I\} + [G]\{\partial I/\partial n\} - \{K\}$

### **3 Le Générateur de Formulations**

#### *3.1 Généralités*

Le générateur est le module qui permet de décrire interactivement une formulation, c'est à dire un fichier contenant la description formelle d'une équation Intégrale de Frontières (IF) qui sera lu ultérieurement par le module de résolution permettant de construire et de résoudre un problème IF. On distingue quatre grandes étapes dans la description formelle d'une équation à l'aide du générateur :

- La description de l'inconnue et des propriétés.
- La description des discrétisations autorisées de l'inconnue.
- La description des intégrales.
- La description des systèmes matriciels.

#### *3.2 La description de l'inconnue et des propriétés*

##### L'inconnue

Dans une formulation matricielle, l'inconnue apparaît sous la forme de plusieurs vecteurs que viennent multiplier les matrices. Ces vecteurs ont plusieurs composantes, chaque composante est une inconnue scalaire du problème à résoudre. Dans un problème de mécanique

des fluides 2D en régime permanent, il y aurait ainsi deux vecteurs de trois composantes chacun :  $I(v_x, v_y, p)$  et  $\partial I / \partial n (\partial v_x / \partial n, \partial v_y / \partial n, \partial p / \partial n)$ . Les composantes des vecteurs sont choisies dans le dictionnaire des variables utilisables, aussi appelé Base de données des Identificateurs. Comme on l'a vu, cette base de données est commune à toutes les équations quelle que soit la méthode utilisée (IF, EF) et facilite ainsi la réalisation de divers couplages. Ce dictionnaire des variables n'est pas limitatif car le concepteur de l'équation peut l'enrichir. Si l'équation doit traiter des cas transitoires, il est probable qu'il apparaisse dans la formulation matricielle des termes dus au traitement du temps; par exemple  $I_0$  (Vecteur  $I$  à l'instant  $t-1$ ). On devra donc décrire autant de déclinaisons de chaque vecteur qu'il y a d'échantillons temporels dans la formulation, de même pour les échantillons d'espace nécessaires pour une éventuelle formulation non-linéaire.

La description d'un vecteur consiste donc à donner son nom et le nombre de composantes qui le forment. Pour chaque composante, on indique aussi son nom, la composante liée ( $\partial v_x / \partial n$  est liée à  $v_x$ ), et si elle est continue ou pas.

### Les propriétés

Les propriétés sont les variables ou les constantes physiques qui apparaissent dans la formulation et qui ne sont pas des inconnues. On distingue deux types de propriétés, les internes et les externes.

Les propriétés internes sont complètement décrites dans l'équation par le Concepteur (Constantes physiques par exemple), leur description est indépendante du problème réel résolu. Les propriétés externes nécessitent la donnée d'un modèle de propriété au moment de la définition du problème réel (Cf 5 dans ce chapitre) par l'utilisateur de l'équation.

La description d'une propriété consiste donc à donner son nom (pris aussi dans le dictionnaire des variables), son type (interne/externe), son nombre de composantes, sa dimension (c'est à dire la dimension des éléments sur lesquels elle est calculable : ponctuelle, linéique, surfacique, volumique), et les types de calculs interdits ou autorisés (Constant, Fonction, Tabulé, Post-processé). Si la propriété est interne, sa description précise est complétée (en donnant la valeur si c'est une constante, par exemple).

Remarque : On appelle caractéristique une composante de l'inconnue ou une propriété.

### *3.3 La description des discrétisations autorisées de l'inconnue*

Le but de cette étape est de préciser la manière dont l'inconnue peut être discrétisée, c'est à dire comment les nœuds vont être répartis sur les éléments qui découpent la géométrie du problème réel. Il s'agit tout d'abord de choisir parmi les éléments de référence de la base de données ceux qui sont autorisés pour cette équation. Les éléments de référence ainsi choisis pourront indifféremment servir de support à la discrétisation numérique, géométrique, et la discrétisation de la formulation (ID. la répartition préliminaire des nœuds de collocation avant

un éventuel déplacement). Les éléments de référence sont regroupés par famille topologique (segments, quadrilatères, tétraèdres, ...). Les éléments de référence suffisent à déterminer complètement les discrétisations numériques (répartition des nœuds) et géométriques (répartition des points), c'est à dire les discrétisation élémentaires. Le concepteur est d'ailleurs amené à préciser, par composante, et par famille topologique, l'élément qui servira de support à la discrétisation numérique. On verra en 4.5 que ce choix peut être modulé lors de la définition de la géométrie discrétisée d'un problème réel.

En ce qui concerne la répartition des nœuds associés aux équations (appelés aussi nœuds de collocation), elle fait partie des attributs des systèmes de l'équation (Cf 3.5) car elle en est une des caractéristiques. On se reportera donc à la description des systèmes de l'équation.

### 3.4 La description des intégrales.

Une intégrale, au sens de la description formelle de la formulation, est une relation entre une équation et des nœuds, destinée à être assemblée, soit dans une matrice, soit dans un vecteur. On a distingué quatre types d'intégrales.

- "Élémentaire matriciel (ELM)" : C'est une relation entre une équation et les nœuds d'un élément où une composante de l'inconnue intervient implicitement. Elle donne donc à assembler un vecteur élémentaire dans une matrice. Sa forme générale est :

$$\int_{\text{élément}} N_i f \alpha_j$$

où :  $N_i$  est un Noyau singulier (i caractérisant le pôle)  
 $f$  une fonction quelconque à calculer sur un élément de frontière  
 $\alpha_j$  un polynome d'interpolation numérique.

- "Élémentaire vectoriel (ELV)" : C'est une relation entre une équation et les nœuds d'un élément où, si l'inconnue intervient, elle est explicite. Elle donne donc à assembler un nombre dans un vecteur. Sa forme générale est :

$$\int_{\text{élément}} N_i f$$

où :  $N_i$  est un Noyau singulier (i caractérisant le pôle)  
 $f$  une fonction quelconque à calculer sur un élément de domaine.

- "Équation matriciel (EQM)" : C'est une relation entre plusieurs nœuds, ces nœuds étant déterminés, soit par l'utilisateur dans l'opérateur de création d'équation (Intégrale singulière), soit automatiquement par le solveur selon le contexte (Coefficient géométrique de collocation). Elle donne donc à assembler un vecteur élémentaire dans une matrice. Il n'y a pas de forme

générale mais pour fixer les idées, ce type d'intégrale sert à assembler dans une matrice des relations du type :  $k_1 \frac{\partial I_1}{\partial n} = k_2 \frac{\partial I_2}{\partial n}$  ou  $\frac{\pi}{2} I$ .

- “Équation vectoriel (EQV)” : C’est un nombre qui ne dépend que de l’équation où il est calculé. Cela donne donc à assembler un nombre dans un vecteur. Ce type d’intégrale sert par exemple à assembler un coefficient de collocation dans un vecteur.

Les deux premiers types d’intégrales (ELM, ELV) sont “à logique d’éléments”. Elles sont le résultat d’une intégration sur un élément et servent donc à calculer toutes les intégrales, “au sens classique du terme”. Les deux autres (EQM, EQV) sont “à logique d’équations”. Ces intégrales servent à calculer les coefficients de collocation, les relations singulières et interfaciales.

Une intégrale, quel que soit son type, est calculée par un ou plusieurs pilotes de traitement. On dispose actuellement de six pilotes (Analytique matriciel-ELM-, Collocation-EQM/EQV-, Numérique matriciel-ELM-, Singulier-EQM-, Numérique vectoriel-ELV, Analytique vectoriel-ELV). C’est l’opérateur critère de traitement qui détermine le nombre de traitements et qui calcule les critères qui aiguillent sur le bon pilote en fonction du contexte.

Une description d’intégrale consiste donc, après avoir donné son nom, son type, et sa dimension éventuelle, à puiser dans la base de données les opérateurs et/ou les objets nécessaires aux pilotes de traitement qui peuvent calculer cette intégrale.

### 3.5 Les systèmes de l’équation

Comme on l’a vu au chapitre 1, la formule qui sert à calculer les valeurs des inconnues à l’intérieur d’un domaine résolu par la MIF n’est pas exactement la même que celle qui a servi à résoudre le système au frontière. De plus, il peut être utile de calculer la valeur de l’inconnue en des points de la frontière qui n’ont pas servi de support à la discrétisation numérique. Ces deux raisons expliquent la nécessité de deux types de systèmes :

- Les systèmes destinés à la résolution
- Les systèmes destinés au Post-Processing

Les deux systèmes se décrivent de la même manière, ils disposent de quatre attributs que le concepteur de la formulation doit choisir :

- l’opérateur de création des équations
- l’opérateur de déplacement des équations
- le vecteur de référence
- le vecteur annexe



Le rôle des opérateurs de création et de déplacement des équations a été vu précédemment. Ils sont à choisir dans la Base de Données. Ils servent à caractériser la manière dont les équations vont être déterminées pour le système décrit.

Les deux autres attributs sont des noms de vecteur, ils sont à choisir parmi les vecteurs de la formulation. Ils servent à déterminer les composantes de l'inconnue qui vont participer à la création des équations. La distinction entre vecteur de référence et vecteur annexe est une information dont le concepteur peut éventuellement se servir dans l'opérateur de création des équations. Typiquement, pour une formulation de la thermique avec les deux vecteurs I (une composante : T) et  $\frac{\partial I}{\partial n}$  (une composante :  $\frac{\partial T}{\partial n}$ ), le vecteur de référence sera I et le vecteur annexe, si nécessaire, sera  $\frac{\partial I}{\partial n}$ .

De plus, comme on va le voir par la suite, la donnée de ces deux vecteurs sert aussi à déterminer le nombre de blocs verticaux de la description matricielle du système.

Remarque : Les deux seuls attributs obligatoires sont l'opérateur de création des équations et le vecteur de référence. On a donc la possibilité de répondre "aucun" pour l'opérateur de déplacement des équations (si la formulation ne traite pas les singularités, par exemple), et pour le vecteur annexe.

### 3.6 La description matricielle des systèmes

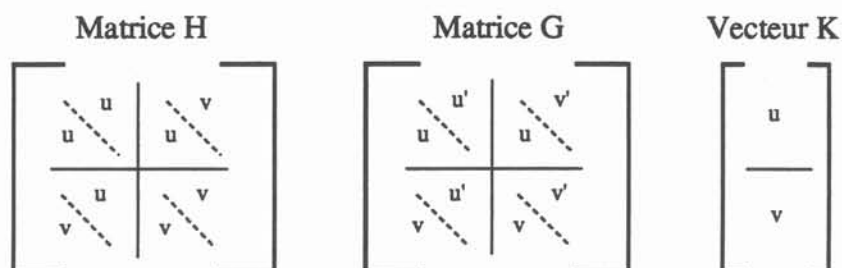
L'opérateur construction de système reçoit en entrée un certain nombre de matrices et de vecteurs globaux que l'on va décrire formellement et est chargé d'en construire d'autres prêts pour l'application des conditions aux limites.

La description formelle de ces vecteurs et matrices est basée sur une vision "bloc" du système matriciel. Chaque bloc d'un tableau est relatif à une composante de l'inconnue de la formulation. Pour un vecteur, il y a autant de blocs verticaux qu'il y a de composantes dans un vecteur fictif composé du vecteur de référence et du vecteur annexe. Pour une matrice, il y a autant de blocs verticaux que pour un vecteur et autant de blocs horizontaux qu'il y a de composantes dans le vecteur qui doit multiplier la matrice. Ainsi, la [figure 3.15](#) schématise la vision "bloc" d'un système à deux matrices : [H], multiplié par le vecteur {I} à deux composantes (u,v) et [G], multiplié par le vecteur {I'} à deux composantes aussi (u',v'). Le vecteur de référence est {I} (il n'y a pas de vecteur annexe).

Une fois le système choisi, il ne reste plus qu'à décrire les tableaux, c'est dire indiquer, bloc par bloc, quels sont les intégrales qui vont y être assemblées. Pour cela, on va donner tout d'abord le nombre d'intégrales du blocs, puis, pour chacune, son nom et le nom d'un modèle

multiplicatif. Le modèle multiplicatif sert à élaborer la combinaison linéaire des intégrales qui vont être assemblées dans le bloc, par exemple une fois le première moins deux fois la seconde. Les modèles multiplicatifs sont donc des modèles constants choisis dans la Base de Données des modèles.

### Vision "bloc" d'un système matriciel



*Les blocs contiennent les noms des composantes de l'inconnue auxquels ils font référence*

**Figure 3.15**

### 3.7 Le Post-Processing

Le Post-Processing de composantes de l'inconnue principale est pris en charge par le système "de Post-Processing". Il se charge de calculer l'inconnue hors de la frontière par une formule décrite par l'utilisateur à l'aide de ce système. Il permet aussi de calculer l'inconnue, aux points de la frontière qui n'étaient pas des nœuds, à l'aide de formules moyennées qui font aussi partie de la description formelle du système de Post-Processing.

Cette formalisation du Post-Processing a un double intérêt : elle permet de dynamiser la manière de calculer l'inconnue aux points qui ne font pas partie de la discrétisation numérique mais surtout elle permet de travailler avec des maillages quelconques et d'obtenir directement la solution en tous les points de ce maillage afin d'effectuer l'exploitation sur un Post-Procasseur du commerce.

## 4 La Description des Discrétisations

### 4.1 Généralités

Comme on peut le voir sur la figure 3.1, le fichier "Discrétisation" (.DIS) est construit à partir d'un fichier "Maillage générique" (.MAI) et d'un fichier "Formulation" (.FRM). Il convient tout d'abord de définir correctement ce que nous avons appelé le Maillage générique : Il s'agit d'une collection d'éléments. Chaque élément réel de cette collection doit être identifié par rapport à un élément de référence de la Base de Données. De plus, les éléments sont

organisés en groupes appelés "Regions de Bases". En d'autres termes, un maillage générique est un ensemble d'éléments qui définissent tous les points et nœuds possibles d'une géométrie.

Un fichier Discrétisation est en fait un Maillage générique adapté aux besoin de Green-Expert et à une formulation spécifique, ou plus précisément à la famille des Éléments de Référence autorisés pour une formulation. En effet, chaque élément origine du maillage générique se retrouvera dans la discrétisation "décliné" en autant d'éléments de Référence autorisés pour cette famille topologique dans la formulation (Cf 3.3). Imaginons par exemple un élément linéique quadratique (à 3 nœuds) issu du maillage générique. Si la formulation autorise l'élément de référence quadratique, mais aussi l'élément de référence linéique linéaire (à 2 nœuds) et constant (à 1 nœud), alors l'élément d'origine du maillage générique donnera naissance à trois éléments dans la discrétisation.

La Description d'une Discrétisation comporte quatre étapes :

- La description des Régions
- La description des Sous-Domains
- La description des Transformations Géométriques
- La descriptions des éléments numériques et de collocation.

### Répartition de Régions de Dicrétisation

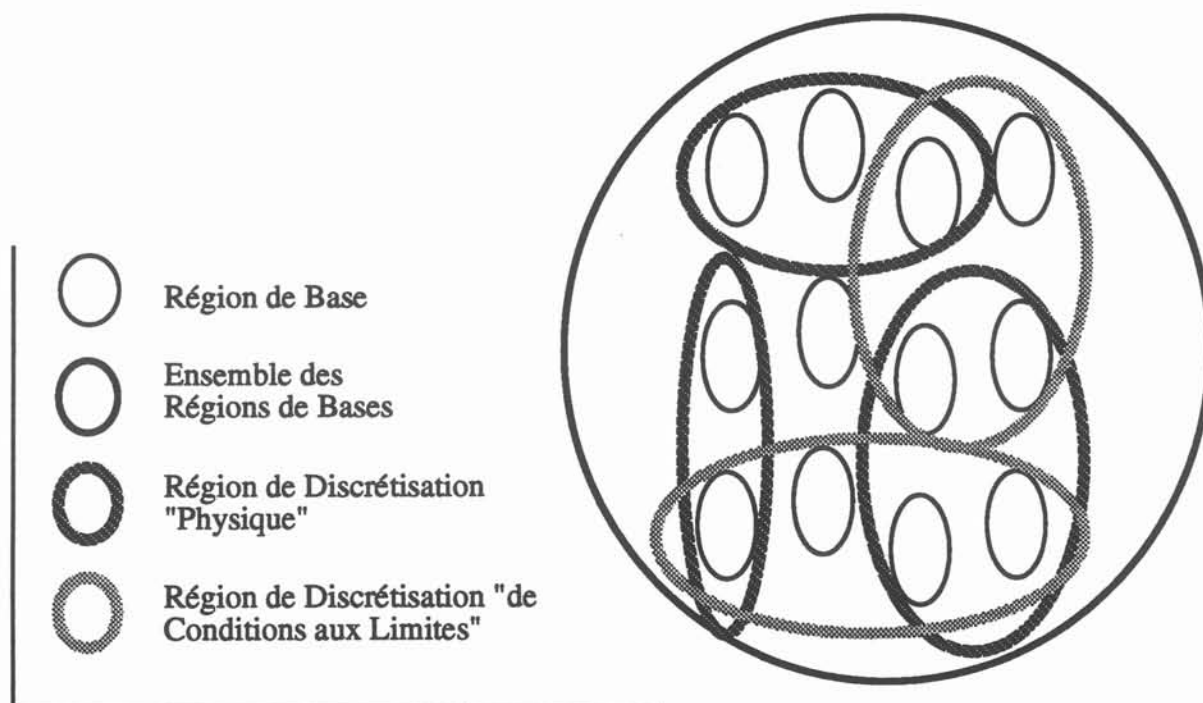


Figure 3.16

## *4.2 Description des Régions de la Discrétisation*

Une "Région de Discrétisation" est un ensemble de Régions de base issues du maillage générique; il y a deux types de régions différents : les "Régions Physiques" et les "Régions de Conditions aux Limites". Comme leur nom l'indique, les premières sont destinées à l'affectation de propriétés physiques tandis que les secondes seront support des conditions aux limites lors de la création d'un problème réel (Cf 5). Il est important de remarquer que les Régions Physiques comme les Régions de Conditions aux Limites forment chacun "un ensemble de sous-ensembles disjoints de Régions de Bases" comme le montre la figure 3.16.

## *4.3 Description des Sous-domaines*

La notion de sous-domaines, qui a été introduite au §1, est primordiale dans les formulations IF puisque le sous-domaine est le support de cette formulation. Plus précisément, on dispose de deux types de sous-domaines :

- Les sous-domaines classiques : La formulation sera intégrée et assemblée sur chacun d'eux séparément. Ils sont formés par un ensemble de Régions de Discrétisation physiques et de conditions aux limites. Ils sont indépendants entre eux et toutes les sortes d'intégrales sont calculables dans ces sous-domaines.

- Les sous-domaines interfaciaux : Ce sont les sous-domaines qui servent à écrire les relations de continuités (ou de discontinuités) entre les sous-domaines classiques. Ils sont aussi formés par un ensemble de Régions de Discrétisation mais seules des intégrales dites "singulières" sont calculables sur ces sous-domaines. On rappelle en effet que ce sont les opérateurs singuliers qui sont destinés au calcul des relations interfaciales.

## *4.4 La description des transformations géométriques*

Contrairement à la méthode des éléments finis, la méthodes des intégrales de frontières n'est pas une méthode locale. Cela implique que seul le calcul d'une formulation sur la totalité d'un domaine d'application (un sous-domaine dans notre terminologie) permet théoriquement d'obtenir le résultat escompté. Or, il est fréquent que, pour des raisons physiques plus ou moins évidentes, la connaissance de la solution sur une partie du domaine suffise à reconstruire exactement la solution sur la totalité du domaine. Il serait gênant d'obliger l'utilisateur à décrire la totalité d'une géométrie, surtout si elle est complexe, à cause de contraintes d'application dues à la méthode alors que, physiquement, cela n'est pas nécessaire.

C'est pourquoi nous avons prévu la possibilité de déclarer formellement des transformations géométriques. Le principe est le suivant :

Le maillage générique ne doit contenir que la partie de la géométrie sur laquelle la connaissance de la solution est suffisante. Prenons l'exemple d'un rectangle disposant de deux

axes de symétrie vis à vis de la solution escomptée, seule la solution sur un quart de rectangle est nécessaire comme le schématise la figure 3.17.

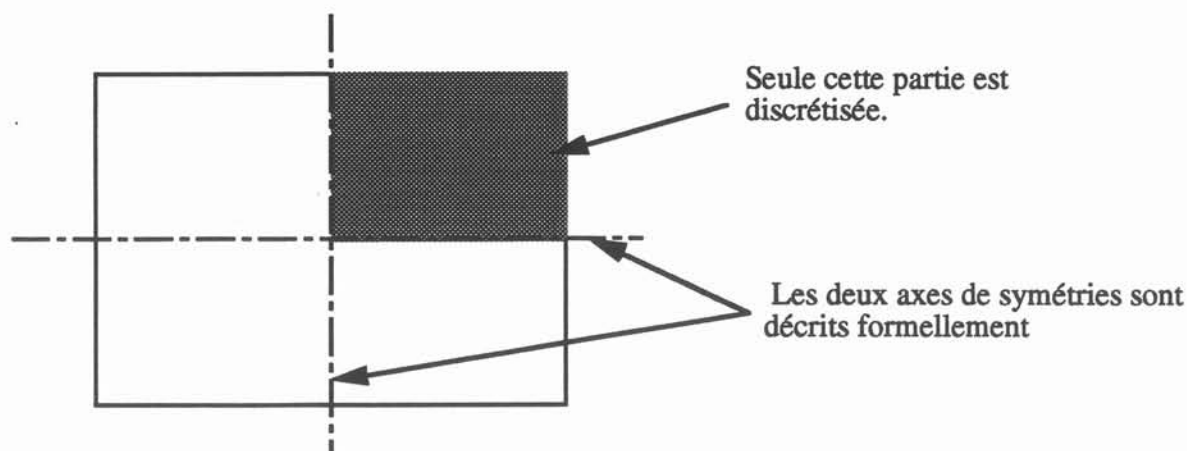


Figure 3.17

La description des transformations géométriques (la plupart du temps des symétries) qui permettent de reconstituer le domaine utilise le principe des matrices de transformations. En effet, on sait que toute transformation géométrique est décomposable en une matrice et un vecteur à appliquer aux points à transformer : Une matrice représentative d'une rotation et un vecteur représentatif d'une translation. Ainsi, la description des transformations géométriques d'une discrétisation consiste à donner ces tableaux pour chacune d'entre elles.

A titre d'exemple, les tableaux correspondants à une symétrie par rapport à une droite 2D d'équation  $ay+bx+c = 0$  sont :

$$\begin{bmatrix} \frac{a^2-b^2}{a^2+b^2} & \frac{-2ab}{a^2+b^2} \\ \frac{b^2-a^2}{a^2+b^2} & \frac{-2ac}{a^2+b^2} \end{bmatrix}$$

pour la matrice et

$$\begin{bmatrix} \frac{-2cb}{a^2+b^2} \\ \frac{-2ca}{a^2+b^2} \end{bmatrix}$$

pour le vecteur.

En plus de ces tableaux, que l'on peut éventuellement décomposer en plusieurs couples si la transformation s'y prête, on aura à fournir deux autres informations qui concernent la nature numérique des transformations. En effet, la brique de base des discrétisations est l'élément et non pas le point. Or, la transformation s'applique à un point pris hors de tout contexte, et notamment hors de son contexte élémentaire. Il faut donc préciser comment la transformation affecte un élément, il y a deux cas : soit l'élément est "réorienté", soit il ne l'est pas. Par

exemple, dans le cas d'une symétrie par rapport à une droite en 2D, un élément linéique est réorienté comme le montre la figure 3.18.

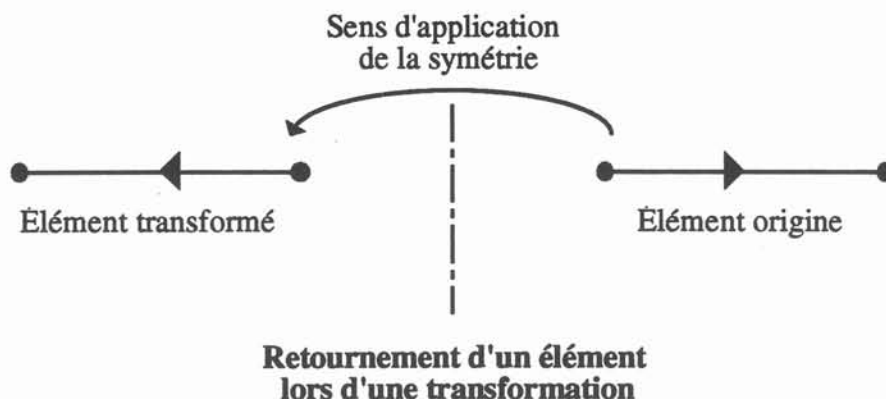


Figure 3.18

La deuxième information à fournir est dépendante du sous-domaine. En effet, tous les sous-domaines ne sont pas forcément concernés par toutes les transformations. Imaginons par exemple la géométrie réelle de la figure 3.19 où les numéros représentent les sous-domaines dans lesquels il faut résoudre. Les deux axes de symétrie du problème nous permettent de ne résoudre que dans un quart de la géométrie, prenons par exemple le quart en haut à droite. Ces deux axes génèrent en fait trois transformations géométriques : Sym1, la symétrie par rapport à l'axe1, Sym2, la symétrie par rapport à l'axe2, et Sym12, la composée des deux précédentes symétries.

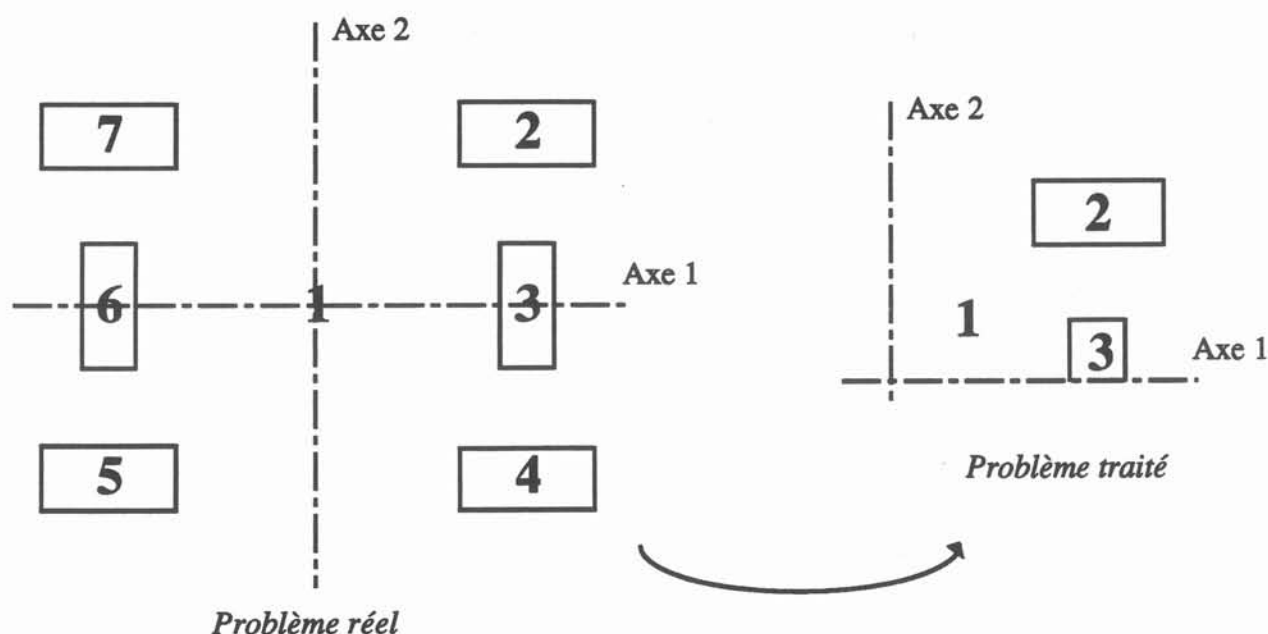


Figure 3.19



Il s'agit donc de décider, pour chaque sous-domaine de la discrétisation effectivement utilisée (figure 3.17), pour chaque transformation, si celle-ci est nécessaire à la résolution ou pas. En ce qui concerne le sous-domaine 2, il est clair que toutes les transformations sont inutiles puisque sa frontière est complètement contenue dans la discrétisation utilisée. Pour le sous-domaine 3, seule la transformation Sym1 est utile puisqu'elle suffit à reconstruire la totalité de sa frontière. Enfin, pour le sous-domaine 1, la frontière a besoin des trois transformations pour être totalement recomposée.

Concrètement, lors de la description d'une transformations géométrique, il est demandé, pour chaque sous-domaine, si la transformation est utile ou non.

#### *4.5 Descriptions des éléments numériques et de collocation.*

Comme on l'a vu en 3.3, le concepteur d'une formulation prescrit la manière dont les nœuds sont répartis sur les éléments en choisissant, par composante de l'inconnue, et par famille topologique autorisée, un élément de référence, que l'on a appelé "élément numérique". Il s'agit ici de préciser ce choix en y rajoutant la notion de région de discrétisation. En effet, il n'est pas impossible de vouloir une discrétisation numérique non homogène, spécialement dans les cas multi-domaines. L'utilisateur est alors amené à choisir parmi les éléments de référence autorisés, un élément numérique, par composante de l'inconnue, par famille topologique, et par région de discrétisation, la réponse par défaut étant bien sûr l'élément de Référence donné dans la formulation pour cette famille topologique et cette composante.

C'est en utilisant la notion d'élément de référence que l'on va décider de la discrétisation initiale de la formulation, ces éléments étant appelés éléments de collocation. A priori, il est logique de penser que les éléments de collocation sont identiques aux éléments numériques et c'est d'ailleurs le choix par défaut qui est proposé à l'utilisateur. Si l'on veut intervenir sur ces choix par défaut, les questions sont identiques à celles posées lors du choix des éléments numériques.

## **5 La Définition d'un Problème IF**

La définition d'un problème de Green-Expert est la dernière étape de la construction du fichier destiné à être résolu par le solveur généralisé, Mif. Cela consiste uniquement à affecter les caractéristiques d'une Formulation pour une Discrétisation. L'affectation des caractéristiques se déroule de la façon suivante :

Pour chaque Région de Discrétisation, pour chaque caractéristique, l'utilisateur doit choisir parmi les modèles disponibles dans la Base de Données. Il peut enrichir la BD si le modèle qu'il désire n'existe pas.

Pour les régions dites "physiques", on n'affecte que les propriétés de la Formulation, les modèles servent alors à calculer la valeur de ces propriétés lors du calcul.

Pour les régions dites "de conditions aux limites", on n'affecte que les inconnues de la formulation, les modèles servent alors à calculer les conditions aux limites spatiales et temporelles du problème.

Ainsi, la construction d'un fichier problème est l'assemblage de toutes les étapes décrites au cours de ce chapitre. Néanmoins, toutes ces étapes étant sauvegardées sous la forme de fichiers, tous ces fichiers sont réutilisables et, par exemple, une fois qu'une formulation a été décrite, elle est réutilisable pour toutes les discrétisations compatibles avec cette formulation. Il est cependant plus pratique, d'un point de vue de l'organisation informatique, de sauvegarder un fichier problème avec tous ses constituants comme le montre la [figure 3.20](#). Cela permet d'assurer l'intégrité du fichier quelle que soit l'évolution de ses constituants, et principalement de la Base de Données, puisque, comme on l'a vu en 2.1, une correspondance est faite entre la Base de Données du fichier Problème et la Base de Données utilisée au moment de l'utilisation du fichier.

#### Composition d'un fichier problème issu de Green-Expert

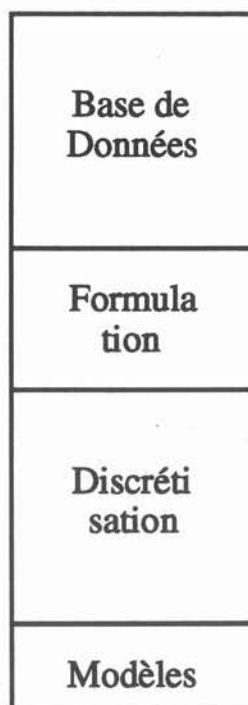


Figure 3.20

## 6 Le Solveur de Green-Expert : Mif

Le principe du solveur Mif est simple : son rôle est de résoudre le problème dont la description est contenue dans un fichier issu de Green-Expert. L'organigramme de la [figure 3.21](#) montre les étapes d'une session du solveur. C'est un découpage grossier que nous détaillerons par la suite mais il permet d'introduire la notion de stratégie de résolution.

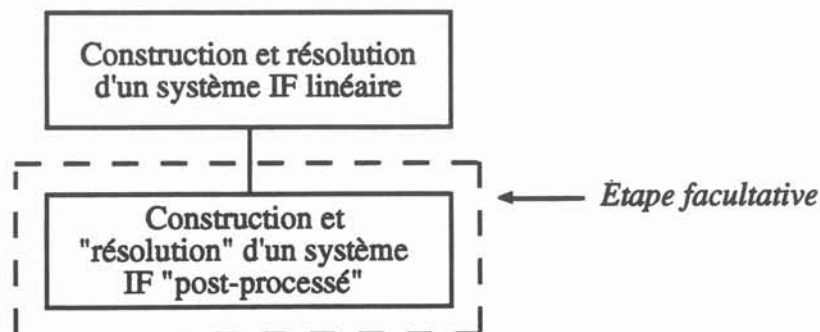
### Déroulement d'une session du solveur



[Figure 3.21](#)

En effet, dans la version actuelle du solveur, seul le cas des formulations linéaires et ne nécessitant pas de schéma temporel de résolution, a été implémenté (il est néanmoins possible de traiter des problèmes transitoires avec des fonctions de Green intégrant le temps). L'aspect algorithmique du traitement des phénomènes non-linéaires, transitoires, où même des phénomènes couplés n'est pas propre à la MIF et nous avons choisi de nous restreindre à un solveur capable de construire tout système linéarisé tout en pensant à un éventuel branchement avec des algorithmes de plus haut niveau pouvant le piloter.

### Stratégie de résolution d'une formulation IF



[Figure 3.22](#)

Ainsi, la figure 3.22 montre les deux étapes de la seule stratégie de résolution implémentée actuellement, celle d'une formulation linéaire disposant d'un système dit "de résolution" et éventuellement d'un système dit "de post-processing". (Cf 3.5)

Nous nous intéressons donc à la construction et à la résolution d'un système IF linéarisé. Nous avons vu lors du chapitre 2 qu'une résolution numérique se ramenait au calcul d'un certain nombre d'intégrales sur un domaine discrétisé. Ces intégrales sont ensuite assemblées dans des matrices ou des vecteurs qui nous permettront de construire un système sur lequel des conditions aux limites seront appliquées. La figure 3.23 montre l'organigramme de ce processus.

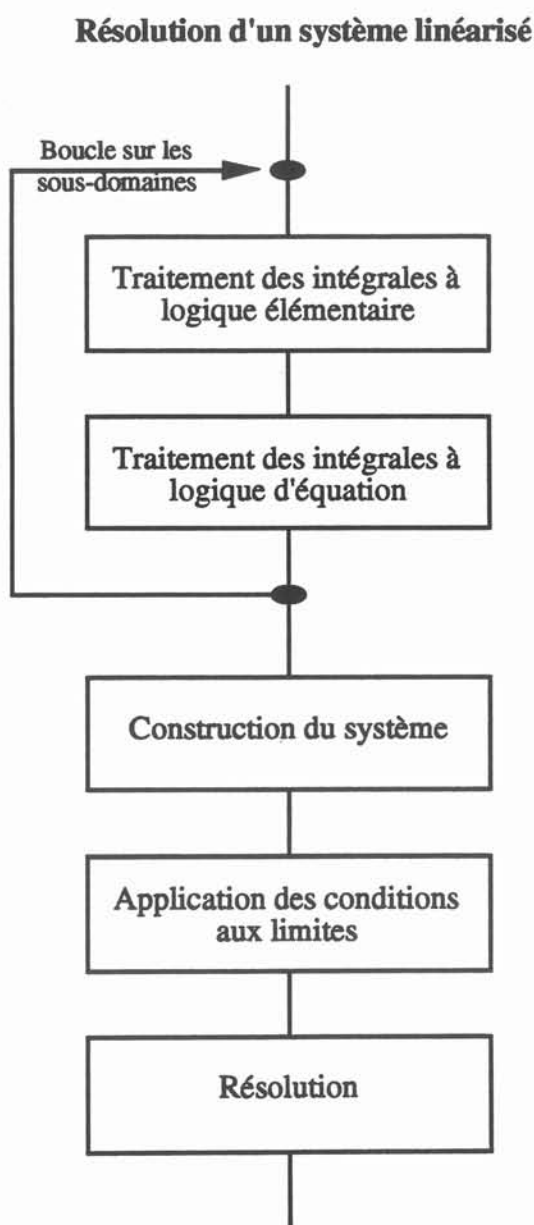


Figure 3.23

Les deux principaux types d'intégrales vus dans la description du générateur font l'objet de traitement séparé car le calcul des intégrales à logique d'éléments nécessite une boucle sur les éléments, non présente pour les intégrales à logique d'équations. C'est d'ailleurs la seule différence d'un point de vue conceptuel entre les deux traitements, c'est pourquoi la [figure 3.24](#) ne schématise que le traitement des intégrales à logique d'éléments.

### Traitement des intégrales à logique d'éléments pour un sous-domaine

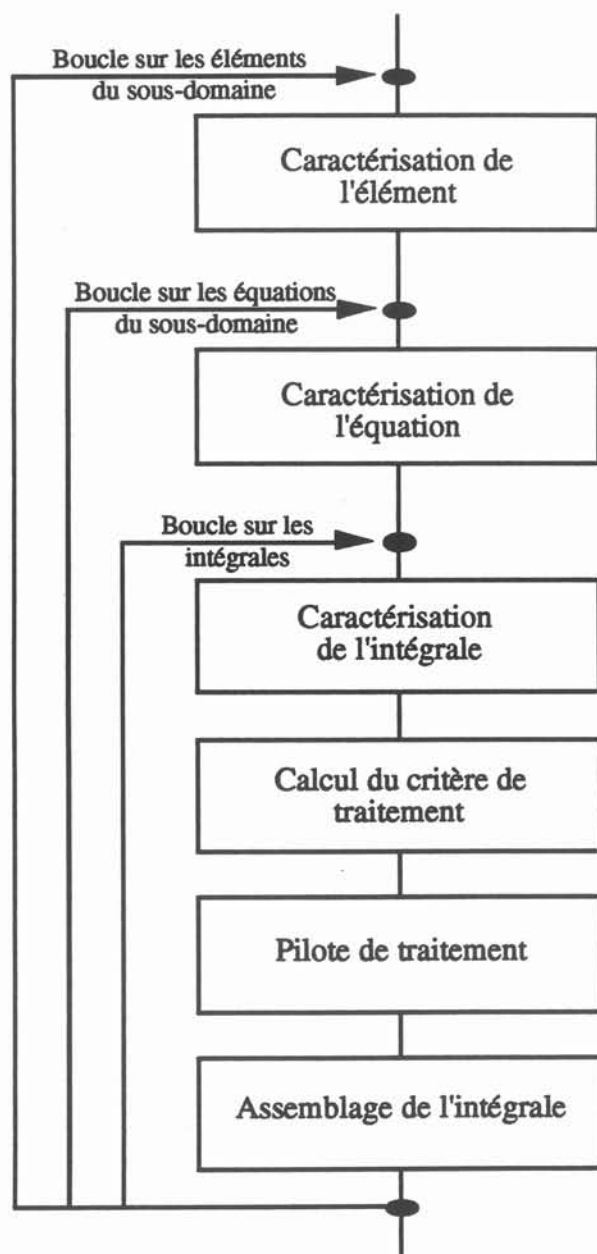


Figure 3.22

La simplicité apparente de la description de ce solveur généralisé ne doit pas faire oublier la complexité des problèmes qu'il est capable de résoudre. En effet, la décomposition analytique de la méthode que nous avons effectuée a permis de matérialiser sous la forme d'opérateurs les étapes d'une formulation IF quelconque, quelle que soit sa complexité. Cependant, les composants de cette formalisation (les opérateurs) doivent être le plus facilement manipulables par un utilisateur non informaticien, il est donc logique de retrouver cette simplicité dans la description du fonctionnement général du solveur Mif.

En fait, là où commencent des calculs propres à une physique donnée, dans un logiciel dédié, nous avons des manipulations de structures générales et réutilisables que sont les opérateurs, dans notre solveur généralisé. Nous avons reporté le travail du physicien au niveau de la Génération de sa Formulation.



### 1 Introduction

---

La concrétisation informatique de notre recherche s'est déroulée en deux phases principales.

La première phase a consisté à prouver la validité et l'intérêt du couplage entre la méthode des éléments finis et des intégrales de frontières dans le cadre d'un système muni d'un Générateur de Formulations; Flux-Expert. La deuxième phase fut principalement consacrée à la Méthode des Intégrales de Frontières et la réalisation de Green-Expert.

L'intégration industrielle de ce projet nous a donc amené à développer des formulations couplées EF-IF en nous servant au maximum des structures logicielles de Flux-Expert, dont les principes généraux sont les mêmes que ceux énoncés au §2. Cependant, ces validations, tout en confirmant la faisabilité du couplage entre les deux méthodes, ont permis d'affiner l'analyse de l'implantation informatique de la Méthode des Intégrales de Frontières dans le cadre que nous nous sommes fixés.

Nous présenterons donc deux exemples caractéristiques de couplage entre les deux méthodes avant quelques exemples de validations du code Green-Expert.

### 2 Couplage EF-IF

---

#### 2.1 Préliminaires

La communauté scientifique s'est très tôt intéressée au couplage entre les méthodes éléments finis et intégrales de frontières [61:ZIENKIEWICZ], car ce couplage permet de profiter des forces de chacune des deux des méthodes tout en évitant leurs faiblesses. Cette complémentarité est résumée dans le tableau suivant. Nous y avons fait figurer des caractéristiques essentielles d'une modélisation numérique en y reportant les comportements classiques des deux méthodes face à ces caractéristiques.

La Méthodes des Intégrales de Frontières (MIF) a été décrite au chapitre 1 de ce mémoire. Quant à la Méthode des Éléments Finis (MEF), rappelons que c'est une méthode basée sur une discrétisation de la totalité du domaine de calcul et non pas de la frontière uniquement. De plus, la formulation est discrétisée localement sur chacun des éléments, contrairement à la MIF où la formulation n'a de sens que sur la totalité d'une frontière. Il existe de nombreux ouvrages

généraux ([16:DHAT], [62:ZIENKIEWICZ] notamment) qui présentent tout à fait didactiquement la méthode.

	<i>MIF</i>	<i>MEF</i>
Limites infinies	<i>Naturelles par l'intermédiaire de la fonction de Green</i>	Possibles [29:IMHOFF] mais contre-nature
Non-linéarités	Possibles mais contre-nature	<i>Naturelles car méthode locale</i>
Frontières libres	<i>Facilitées par l'absence de maillage intérieur</i>	Complicées par la présence d'un maillage intérieur
Hétérogénéités	Les seules qui sont facilement traitables sont celles qui sont homogènes par morceaux	<i>Naturelles car méthode locale</i>
Singularités physiques	<i>Facilitées par une continuité non obligatoire et par la présence naturelle de pôles singuliers</i>	Complicées car obligation d'une continuité inter-élément dans une formulation standard [3:BLUM]
Complexité de la formulation	N'est possible que par la mise au second membre (Sources) de tous les termes ne permettant pas l'application d'un principe de réciprocité	<i>Avec une méthode projective, les seules limitations sont numériques et non au niveau de la formulation</i>

Ce tableau montre que, même s'il existe pour chacune des deux méthodes des techniques autorisant la prise en compte de toutes les difficultés d'une modélisation numérique, leur couplage permet effectivement d'utiliser leurs *points forts* et d'éviter leurs points faibles.

En pratique, dans un problème où l'on veut utiliser une formulation couplée, il existe un domaine  $\Omega_{ef}$  (de frontière  $\partial\Omega_{ef}$ ) où l'on applique la MEF et un domaine  $\Omega_{if}$  (de frontière  $\partial\Omega_{if}$ ) où l'on applique la MIF. Les frontières  $\partial\Omega_{ef}$  et  $\partial\Omega_{if}$  ont une partie commune  $\partial\Omega_{ei}$  sur laquelle le couplage va avoir lieu, c'est à dire que les formulations EF et IF vont partager, sur  $\partial\Omega_{ei}$ , des inconnues. On trouve dans la littérature, logiquement, deux approches de couplage en ce qui concerne la formulation, l'une étant plutôt orientée "éléments finis" et l'autre "intégrales de frontières", la différence entre les deux orientations résidant dans le choix des inconnues qui vont servir au couplage.

Considérons, pour simplifier, qu'une formulation IF utilise la variable d'état  $u$  et sa dérivée normale  $u'$  sur  $\partial\Omega_{if}$  et qu'une formulation EF utilise  $u$  sur  $\Omega_{ef}$  et fait apparaître  $u'$  sur  $\partial\Omega_{ef}$  (Cf. 2.2 ci-après). L'approche orientée "éléments finis" consiste à exprimer, en inversant localement les matrices IF,  $u'$  en fonction de  $u$  sur  $\partial\Omega_{ei}$ , et à reporter alors  $u'$  dans la

formulation EF. Le couplage se réalise sur  $u'$  et cela nécessite une inversion locale des matrices IF. L'approche orientée "intégrales de frontières" consiste à considérer  $u'$  comme une inconnue de la formulation EF. Cela augmente le nombre d'inconnues de la formulation EF mais permet un couplage naturel des deux formulations sur  $\partial\Omega_{ei}$  puisqu'elles utilisent les mêmes inconnues. C'est cette dernière approche que nous avons utilisée car elle nous semble plus générale que la première et plus adaptée à un système disposant d'un Générateur de formulations.

Nous avons mis en œuvre deux exemples de couplages qui exploitent les avantages des deux méthodes, de la Magnéto-thermique avec des non-linéarités et de la Magnétodynamique avec des parties mouvantes. Mais avant d'exposer ces deux exemples, nous expliquons brièvement les techniques employées pour leur implantation.

## 2.2 Techniques employées

Nous nous limiterons aux considérations essentielles de notre implantation du couplage.

Soit  $\Omega_{ef}$  et  $\Omega_{if}$  deux domaines, de frontières  $\partial\Omega_{ef}$  et  $\partial\Omega_{if}$ , dans lesquels on veut résoudre le problème différentiel à l'aide de la MEF et de la MIF :

$$L(u) = 0 \quad (4.1)$$

On a vu (1.9) que, pour la MIF, on obtenait une formulation faible du type suivant :

$$c_1 u_i = \langle S^*(S_i^f), G(u) \rangle_{\partial\Omega_{if}} + \langle G^*(S_i^f), S(u) \rangle_{\partial\Omega_{if}} \quad (4.2)$$

La MEF, quant à elle, permet, dans sa formulation de Galerkin d'obtenir des formulations faible du type :

$$\langle \alpha_j, R(u) \rangle_{\Omega_{ef}} + \langle \alpha_j, \psi(u) \rangle_{\partial\Omega_{ef}} = 0 \quad (4.3)$$

Généralement, dans une formulation éléments finis classique, les intégrales sur la frontière permettent d'exprimer les conditions aux limites de type Neumann et sont donc explicitées afin de faire apparaître des termes sources dans la formulation matricielle résultante. Or, ici, dans l'optique du couplage avec la formulation intégrales de frontières, on va garder au premier membre ces intégrales de frontières. Afin d'effectuer le couplage "orienté IF" décrit dans les préliminaires, il faut que l'on dispose des mêmes expressions de l'inconnue dans les deux formulations. Il est possible que cela nécessite un réarrangement des termes dans la formulation EF. Pour simplifier notre propos, on considérera que l'on a, dans les deux formulations, des intégrales aux frontières dépendant de  $\frac{\partial u}{\partial n}$ , on peut alors écrire les deux formulations matricielles suivantes :

$$\text{MEF} \quad [A] \{u\} + [B] \left\{ \frac{\partial u}{\partial n} \right\} = 0 \quad (4.4)$$

$$\text{MIF} \quad [H] \{u\} + [G] \left\{ \frac{\partial u}{\partial n} \right\} = 0 \quad (4.5)$$

Les matrices [A] et [B] sont creuses et les matrices [H] et [G] sont pleines. Malgré ces différences, nous avons implanté un couplage fort, c'est à dire que les intégrales des deux méthodes sont assemblées dans les mêmes matrices. Le système à résoudre est donc :

$$\text{MEF+MIF} \quad \begin{bmatrix} A & B \\ H & G \end{bmatrix} \begin{Bmatrix} u \\ \frac{\partial u}{\partial n} \end{Bmatrix} = 0 \quad (4.6)$$

Par exemple, pour l'équation de Laplace en 2D, le système matriciel est le suivant :

$$\begin{bmatrix} \int_{\Omega_i} \text{Grad}(\alpha_i) \cdot \text{Grad}(\alpha_j) d(\Omega_i) & - \int_{\partial\Omega_i} \alpha_i \beta_j d(\partial\Omega_i) \\ \frac{1}{2\pi} \int_{\partial\Omega_i} \frac{\partial \ln(M_i M_j)}{\partial n} d(\partial\Omega_i) - \frac{\delta_{ij}}{2} & - \frac{1}{2\pi} \int_{\partial\Omega_i} \ln(M_i M_j) d(\partial\Omega_i) \end{bmatrix} \begin{Bmatrix} u \\ \frac{\partial u}{\partial n} \end{Bmatrix} = 0$$

où la première intégrale de la formulation EF, qui est surfacique, est la forme classique du terme de conduction dans une discrétisation par la MEF. Par contre, la deuxième, linéique, est celle qui n'apparaît pas d'habitude dans une formulation classique car elle est remplacée par des conditions aux limites.

On peut constater que la formulation EF utilise deux types de polynomes ( $\alpha$  et  $\beta$ ). En effet, si la formulation IF est discrétisée à l'aide d'éléments constants, la formulation EF est discrétisée à l'aide d'éléments quadratiques pour  $\{u\}$  et des mêmes éléments constants pour  $\{\frac{\partial u}{\partial n}\}$ . En ce qui concerne l'assemblage, puisque l'on a utilisé les structures logicielles de Flux-Expert, un élément spécial appelé macro-élément est introduit. Cet élément contient tous les nœuds milieux des éléments de la frontière commune entre  $\Omega_{ef}$  et  $\Omega_{if}$ ,  $\partial\Omega_{ei}$ . Puisque l'on a pris des éléments constants pour la formulation IF, ces nœuds sont identifiables, d'une manière bi-univoque, aux éléments de frontières auxquels ils appartiennent. D'autre part, ces éléments constants ont l'avantage d'éviter les singularités physiques (Cf. §1) et permettent aussi l'identification des nœuds avec les équations IF. Ainsi, il est possible dans un premier temps, de calculer les matrices IF [H] et [G] qui contiennent des valeurs qui lient tous les nœuds du macro-élément, et dans un deuxième temps, d'assembler dans les matrices globales, ces deux matrices comme s'il s'agissait de matrices élémentaires classiques. On obtient ainsi l'organigramme de la figure 4.1 sur lequel on a isolé la partie logicielle qui est issue du solveur EF de Flux-Expert.

## Résolution d'un système linéarisé Flux-Expert (MEF) couplé MIF

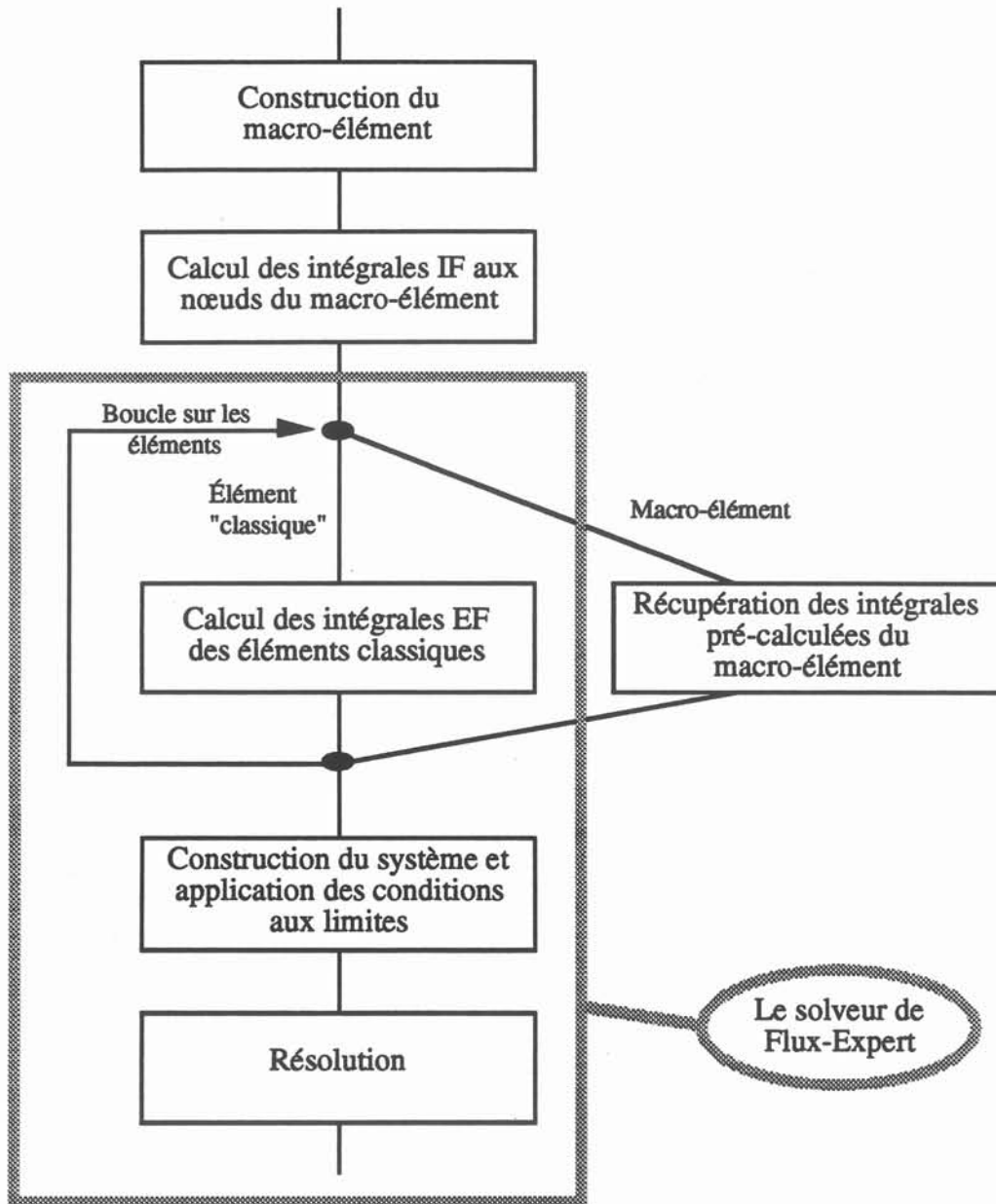


Figure 4.1

### 2.3 Magnéto-thermique non-linéaire

L'intérêt de ce test est de mettre en œuvre deux types de couplages. On couple d'une part le problème magnétique au problème thermique en injectant les pertes joules du premier comme source du deuxième et d'autre part, la MEF et la MIF dans les deux problèmes. Il s'agit de modéliser un problème de chauffage par induction, le creuset ayant en partie des propriétés thermiques non-linéaires. Les équations aux dérivées partielles de départ sont :

Pour la thermique :  $\text{Div} ( -k \text{ Grad} (T) ) = Q$  (4.7)

où : T est la température en Kelvin  
k est la conductivité thermique  
Q les sources thermiques

Pour le magnétique :  $\text{Rot} ( v \text{ Rot} (A) ) = J$  (4.8)

où : A est le potentiel vecteur perpendiculaire au plan de l'étude  
v la susceptibilité magnétique

J les courants dans le plan de l'étude décomposables en courants :

- de transports (  $\sigma V \times \text{Rot} (A) )$  (V : vitesse des pièces en mouvement)
- de Foucault (  $-\sigma \frac{\partial A}{\partial t}$  ) ( $\sigma$  : conductivité électrique)
- d'excitation (  $J_{\text{ex}}$  )
- d'aimantation (  $\text{Rot} (M_r) )$  ( $M_r$  : aimantation)

#### Géométrie du Couplage MEF-MIF en magnéto-thermique

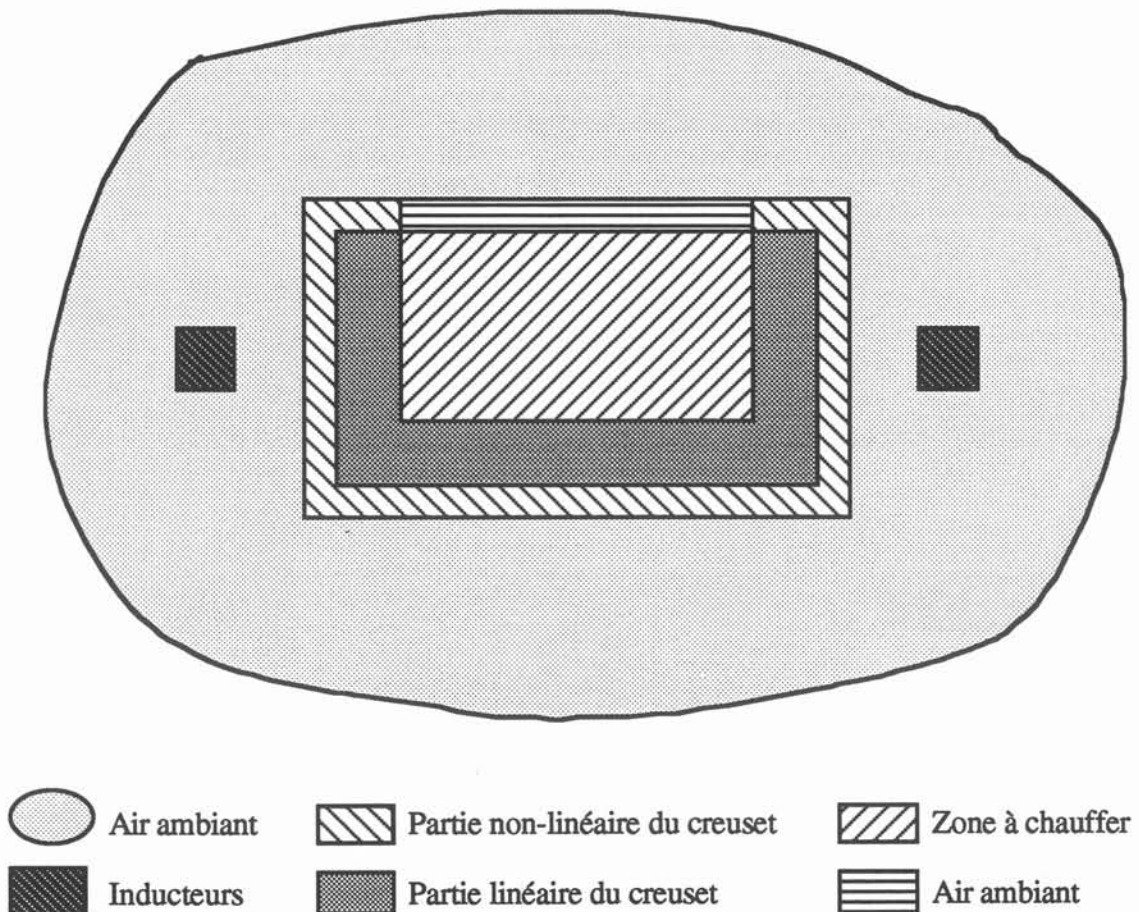



Figure 4.2







La géométrie du problème est donnée sur la figure 4.2.

Pour le problème magnétique, seul l'air ambiant (  ) est résolu avec la MIF, l'intérêt étant bien sûr la modélisation exacte de l'infini. En effet, dans l'air, l'équation (4.8) se simplifie puisqu'il n'y pas de courants et que la susceptibilité magnétique est constante dans l'air : on doit résoudre  $\text{Rot}(\text{Rot}(A)) = 0$ .

Écrivons  $\text{Rot}(\text{Rot}(A))$  en 2D : A n'a qu'une composante perpendiculaire au plan de l'étude, on a donc :

$$\text{Rot}(\text{Rot}(A)) = \begin{pmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} \end{pmatrix} \wedge \begin{pmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} \end{pmatrix} \wedge \begin{pmatrix} 0 \\ 0 \\ A \end{pmatrix} = \begin{pmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} \end{pmatrix} \wedge \begin{pmatrix} \frac{\partial A}{\partial y} \\ -\frac{\partial A}{\partial x} \\ 0 \end{pmatrix} = - \begin{pmatrix} 0 \\ 0 \\ \frac{\partial^2 A}{\partial x^2} + \frac{\partial^2 A}{\partial y^2} \end{pmatrix} \quad (4.9)$$

Ainsi, l'équation vectorielle  $\text{Rot}(\text{Rot}(A)) = 0$  est équivalente, en 2D, à l'équation scalaire  $\Delta A = 0$ . Le traitement de cette équation de Laplace par la MIF a été décrite au chapitre 1. Le principe du couplage effectué avec la MEF est celui exposé précédemment.

La figure 4.3 montre les isovaleurs de A sur la géométrie. On remarquera que, bien que l'on ait limité l'exploitation des résultats à une certaine "boite d'air", cette limitation n'a aucune influence sur l'allure des résultats. En effet, rappelons que la formulation IF est résolue sur la frontière de l'air ambiant (  ), qui comprend les frontières extérieures de zones résolues avec la MEF : les inducteurs (  ), le creuset (  ), de l'air ambiant (  ) mais aussi l'infini. Or, on a vu au cours du chapitre 1 que les intégrales de frontières s'annulaient sur la frontière infinie, ce qui permet bien de résoudre ce problème extérieur en ne maillant que les frontières des objets résolus par la MEF. Ceci implique que l'on peut, à l'issue de la résolution, calculer la valeur de A, n'importe où dans l'air ambiant, la valeur étant exacte même sur la frontière de la "boite d'air" englobante. Ce résultat est remarquable si on le compare à celui obtenu lors d'une résolution d'un tel problème extérieur par la MEF. En effet, avec une méthode des éléments finis classique, on est amené à limiter le domaine de résolution à une "boite d'air". La différence avec la MIF réside dans le fait que, puisque le domaine est maillé, on doit imposer des conditions aux limites aux frontières extérieure de la "boite d'air" qui sont censées modéliser l'infini. En pratique, pour une résolution par la MEF de notre problème magnétique, on aurait imposé  $A=0$  aux frontières extérieures de la "boite d'air", ce qui représente bien sûr une approximation d'autant plus fausse que la boite est petite.

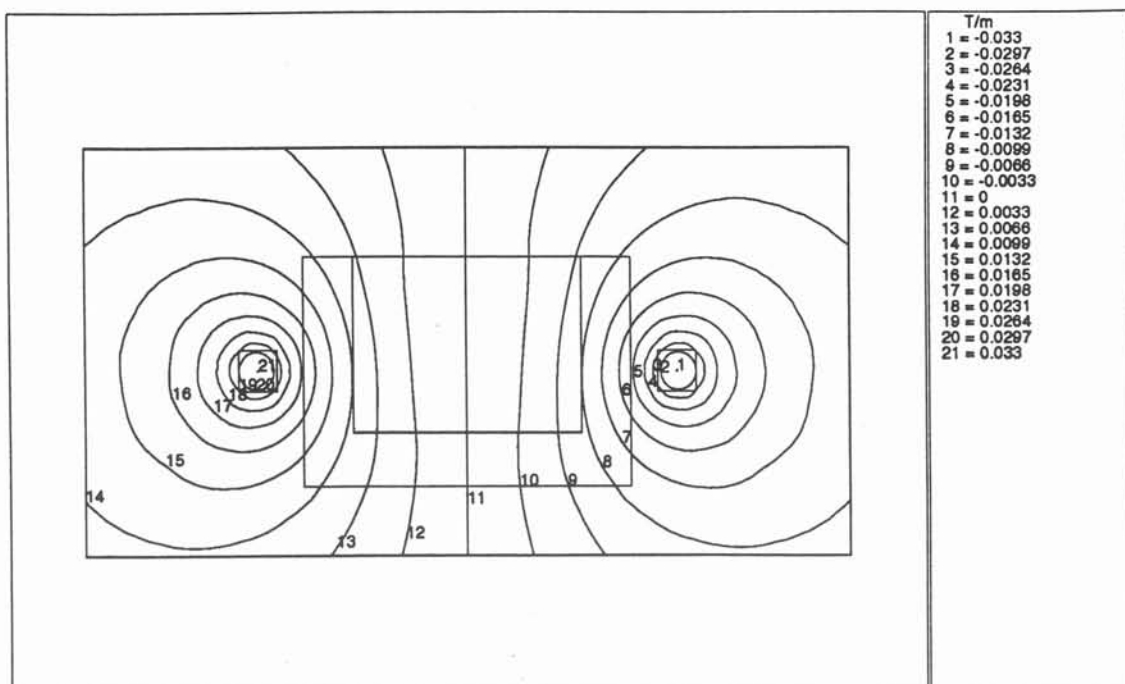


Figure 4.3

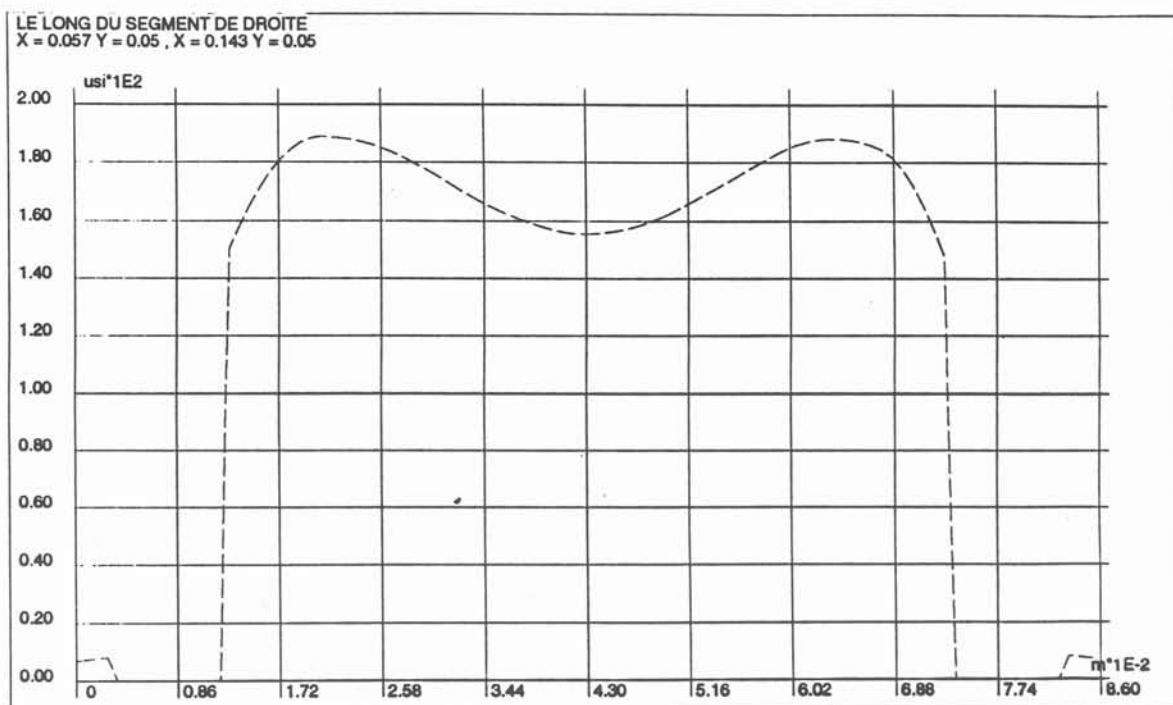





Figure 4.4

Le problème thermique est non-linéaire : on considère que le creuset a une conductivité thermique non-linéaire dans sa zone en contact avec l'air (  ) alors que le reste du creuset

(  ) a un comportement linéaire. Le métal chauffé a aussi une conductivité thermique non linéaire. La figure 4.4 montre la valeur de la conductivité thermique le long d'un segment de droite horizontale au milieu de la géométrie qui est réduite au creuset et au métal pour cette partie thermique. On remarque que les zones où la conductivité est nulle (sur la courbe) correspondent en fait à la partie résolue à l'aide la MIF (  ).

Les sources proviennent du problème magnétique précédent sous la forme de pertes joules ( $\frac{1}{2} \sigma \omega^2 A^2$ ), seul le métal étant supposé conducteur. Les valeurs des pertes ont été sauvegardées dans un fichier lors de l'exploitation du problème magnétique et sont récupérées dans le problème thermique lors de sa définition.

Dans la zone résolue à l'aide de la MIF l'équation  $\text{Div} (-k \text{ Grad } (T)) = Q$  se réduit une fois encore à  $\Delta T = 0$  puisqu'il n'y a pas de sources ( $Q=0$  car le creuset est non conducteur) et  $k$  est constant.

La figure 4.5 montre les isovaleurs de la température pour ce problème.

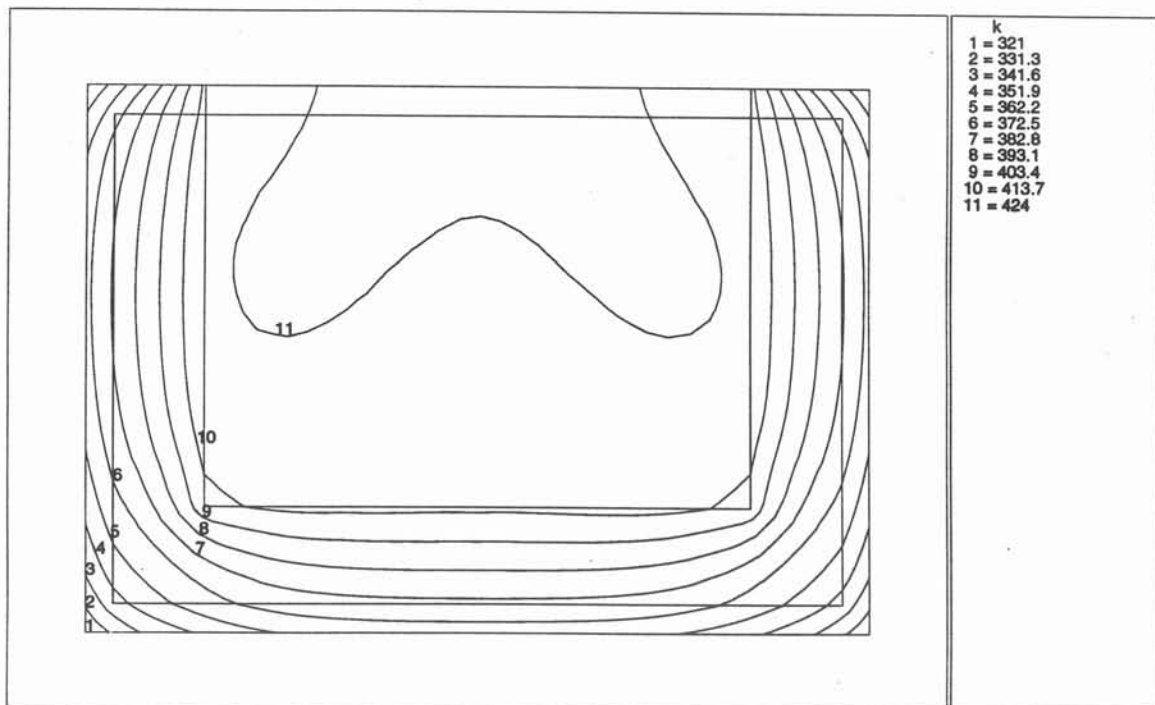
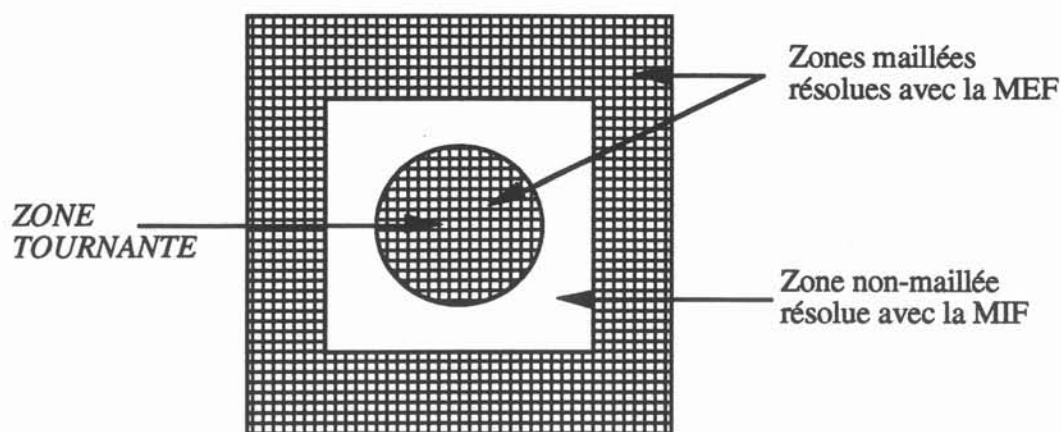


Figure 4.5

#### *2.4 Magnétodynamique avec partie mouvante*

L'électrotechnique, et plus généralement toute l'ingénierie axée autour de l'électromagnétisme, est un des domaines de prédilection des applications du couplage des deux méthodes. Dans le cas des machines tournantes, il est tout à fait intéressant de modéliser

l'entrefer avec la MIF, ce qui permet de faire tourner le Rotor sans problèmes de maillage puisque l'entrefer n'est pas maillé comme le montre le schéma de principe de la figure 4.6.



**Schéma de principe d'un couplage MEF-MIF  
pour modélisation machine tournante**

Figure 4.6

L'exemple de ce type de problème que nous avons implanté est une modélisation 2D simplifiée d'un compte-tours. La géométrie du problème résolu est en figure 4.7. Il s'agit d'un aimant qui provoque un champ canalisé par un noyau magnétique. Les encoches de la partie tournante provoquent des variations de flux qui peuvent être détectées par une bobine de cuivre (en appliquant la loi de Lenz :  $e = -\frac{d\Phi}{dt}$ ).

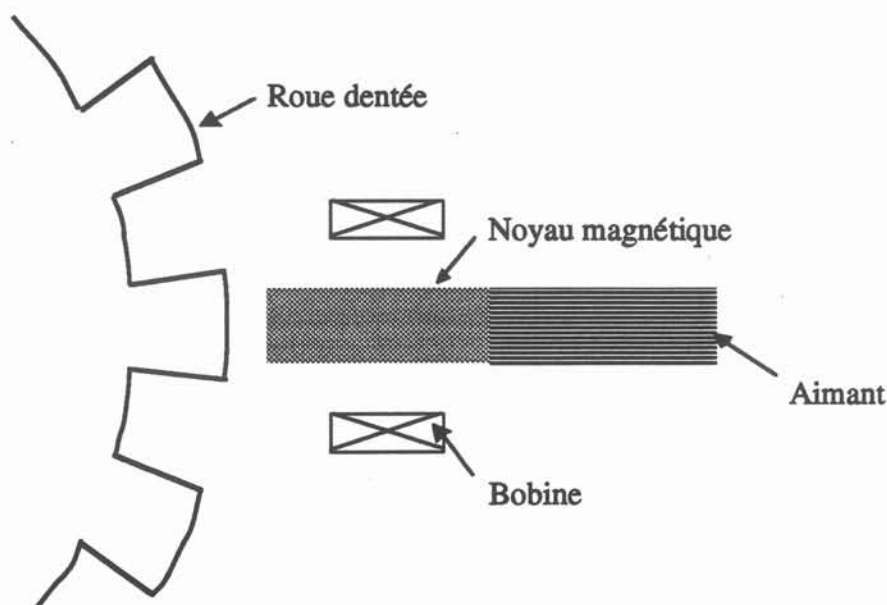
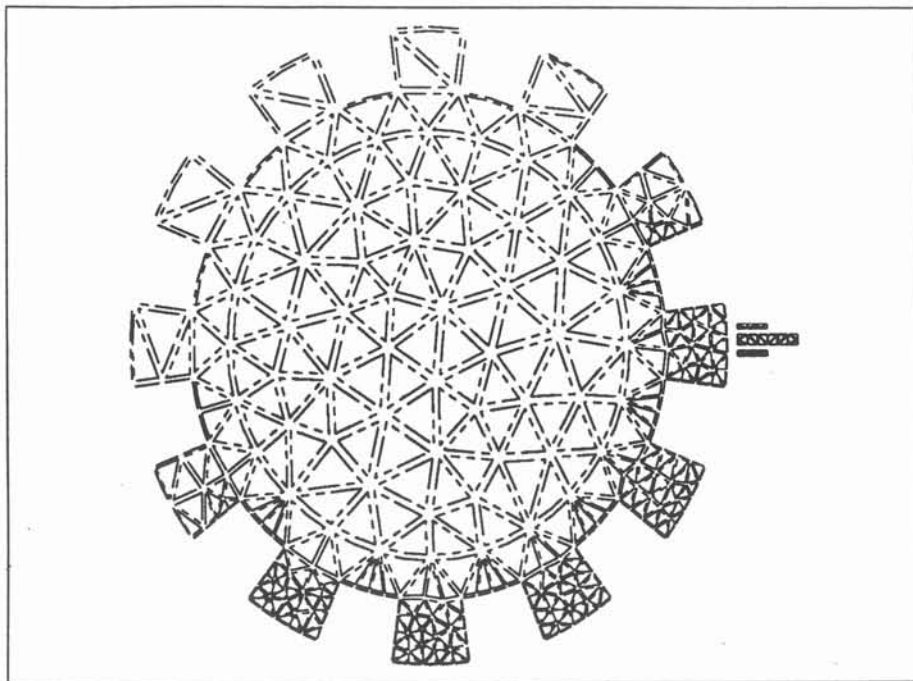


Figure 4.7

L'équation est celle déjà décrite pour la partie magnétique du problème magnéto-thermique (c'est à dire l'équation (4.8) :  $\text{Rot} (\nu \text{ Rot} (A)) = J$ ) mais les propriétés physiques font apparaître d'autres types de courants : des courants de Foucault et des courants dus au déplacement de la roue dentée, une aimantation dans l'aimant. La méthode des intégrales de frontières est utilisée dans l'air et permet donc de faire tourner la roue dentée au cours de la résolution et au cours de l'exploitation comme on peut le constater sur les figures suivantes. Comme pour le problème précédent, l'équation se limite à la résolution de l'équation de Laplace (pour la partie résolue par la MIF) pour la composante perpendiculaire au plan de l'étude du potentiel vecteur (la seule qui soit non nulle dans une modélisation 2D avec un champ magnétique dans un plan) . Contrairement au problème précédent, nous n'avons pas effectué de Post-Processing dans l'air car cela aurait nettement compliqué l'exploitation des résultats avec le Post-Processeur que nous avons utilisé : celui de Flux-Expert. En effet, ce Post-Processeur nécessite la présence d'un maillage dans la zone exploitable. Or, comme nous l'avons expliqué, l'air n'est ici pas maillé pour permettre une rotation aisée de la roue dentée. Le maillage est présenté en figure 4.8.



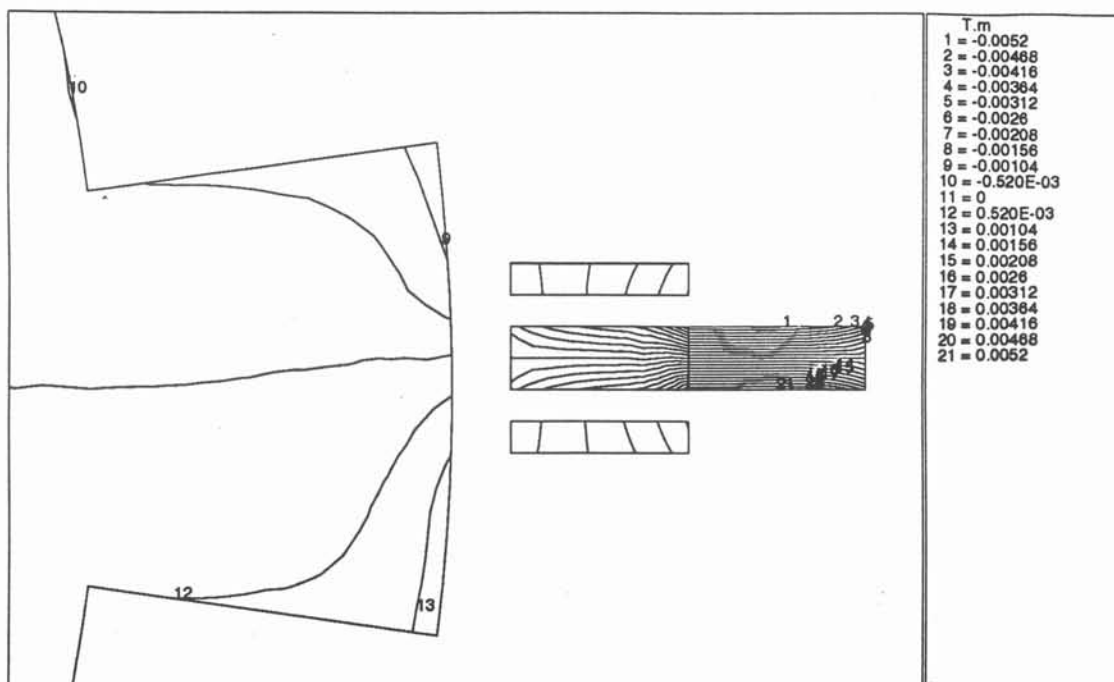
**Maillage**

Figure 4.8

Il convient par ailleurs de préciser que les bobines censées mesurer le courant ne sont en fait pas modélisées, car la modélisation 2D ne serait pas correcte pour notre problème physique, mais que l'on conserve une zone maillée à l'endroit où elles seraient situées afin de pouvoir effectuer l'exploitation dans cette zone. On supposera que les bobines ne perturbent que très faiblement le champ créé par l'aimant, le potentiel vecteur ainsi calculé dans ces petites zones

maillées à la place des bobines sera donc très proche des courants induits qui y seraient mesurés.

La figure 4.9 montre un détail de la géométrie pour la position initiale. On y voit une concentration des isovaleurs du potentiel vecteur dans les parties fixes, ce qui est normal puisque la seule source est l'aimant.

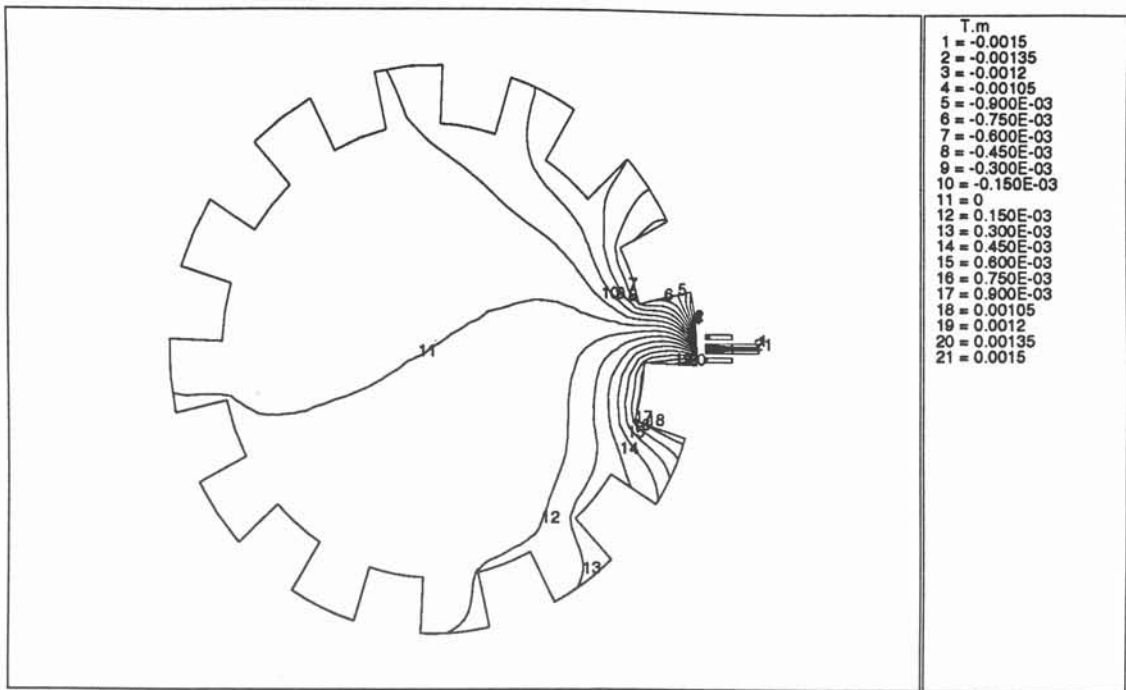


**Isovaleurs de A. Position initiale**

Figure 4.9

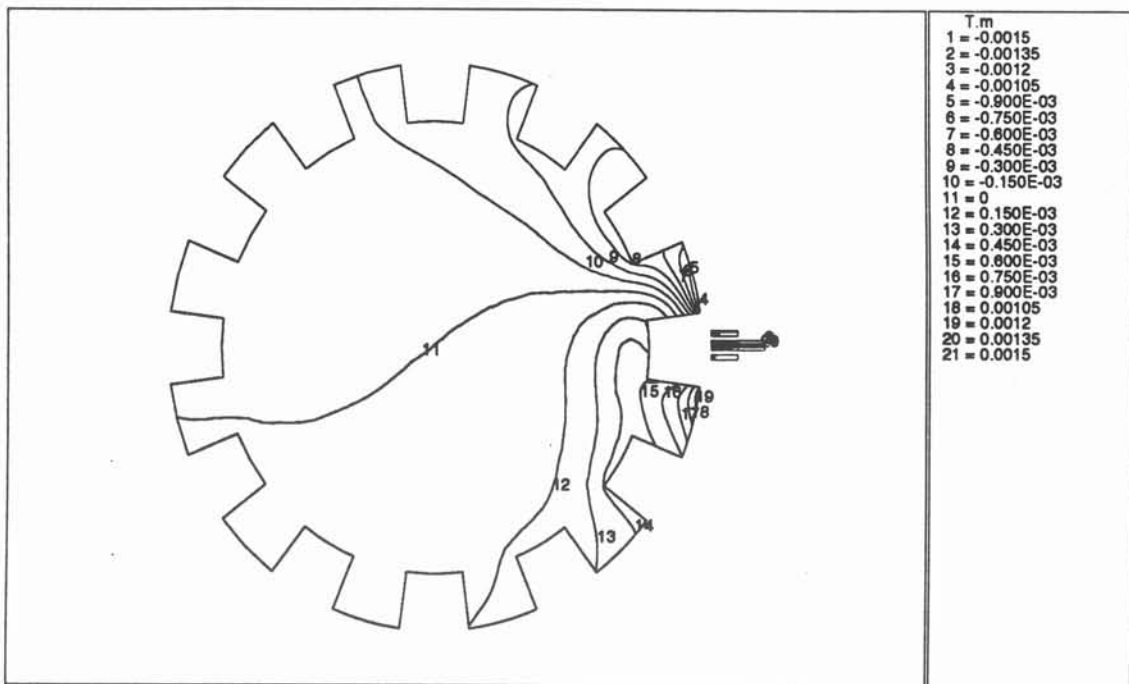
Les figures 4.10 et 4.11 montrent les isovaleurs du potentiel vecteur dans la roue dentée pour deux positions différentes de celle-ci. On y observe clairement l'effet "dissymétrisant" de la rotation de la roue.





Isovaleurs de A. Après rotation 40 degrés

Figure 4.10



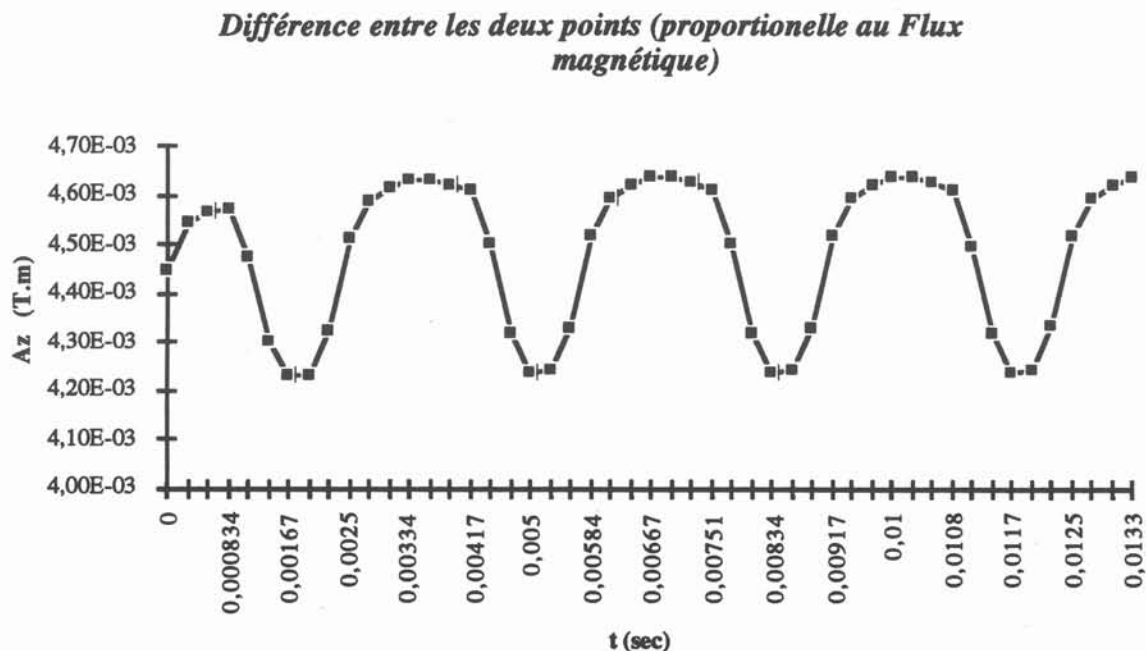
Isovaleurs de A. Après rotation 50 degrés

Figure 4.11

Enfin, la [figure 4.12](#) montre l'évolution temporelle de la différence de potentiel A entre deux points de part et d'autre du noyau magnétique, situé dans les petites zones maillées dans

lesquelles se situeraient les bobines de mesure du compte-tours. Cette différence est intéressante car on peut montrer qu'elle est proportionnelle au flux magnétique  $\Phi$ .

On observe bien sur la courbe de la figure 4.12 une évolution sinusoïdale et l'on en déduit que la différence de potentiel  $e = - \frac{d\Phi}{dt}$  a aussi une allure sinusoïdale, résultat qui est qualitativement conforme à ce que l'on attendait.



**Figure 4.12**

### 3 Validation de Green-Expert

---

#### 3.1 Préliminaires

Nous tenons à souligner qu'il n'est pas dans notre intention ici d'exposer toutes potentialités de notre système mais d'en montrer la validité. Comme nous l'avons déjà expliqué, le système est par nature évolutif et il serait donc fastidieux, dans une phase de recherche, de développer un grand nombre de formulations, puisque c'est de cela qu'il s'agit. Nous nous limiterons donc dans le cadre de ce mémoire à la description de l'utilisation Green-Expert pour graduellement augmenter le champs d'application d'une formulation 2D à l'aide du Générateur de Formulations.

Nous exposerons ensuite un exemple simple de résolution 3D.

#### 3.2 De l'Équation de Laplace à l'équation de Darcy ....

On se propose de générer la formulation correspondant à l'équation de la chaleur dans un espace à deux dimensions :

$$\text{Div}(-k \text{ Grad}(T)) = 0 \quad (4.10)$$

où :  $k$  est la conductibilité thermique (éventuellement tensorielle),  
 $T$  représente l'inconnue.

Nous allons procéder en plusieurs étapes afin de comprendre l'utilisation et l'utilité du Générateur de Formulations. Chaque étape correspond à une simplification plus ou moins grande des conditions d'application de l'équation (4.10). En quatre étapes nous allons passer de l'équation de Laplace applicable sur des discrétisations à l'ordre 0 sans interfaces à l'équation de Darcy applicable sur des discrétisations à l'ordre 1 avec interfaces.

#### Première étape : Laplace ordre 0 sans interfaces

Pour cette première étape, on considère le cas où  $k$  est constant. On a alors déjà vu que l'équation (4.10) se simplifiait et devient :

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = 0 \Leftrightarrow \Delta T = 0 \quad (4.11)$$

Afin de simplifier au maximum notre première formulation, on va résoudre l'équation (4.11) en utilisant une discrétisation numérique à l'ordre 0, c'est à dire que les inconnues de la formulation ( $T$  et  $\frac{\partial T}{\partial n}$ ) sont constantes sur chacun des éléments de la frontière du domaine sur lequel on veut résoudre l'équation, et une discrétisation géométrique à l'ordre 1, c'est à dire que

la frontière est décrite par des segments rectilignes. Or, le traitement de cette équation avec des éléments numériques constants a été vu en détail lors du chapitre 1. On rappelle que l'on obtient la formulation suivante :

$$\forall j, - \sum_{i=1}^N \frac{\partial T_i}{\partial n} \int_{\partial\Omega_i} \ln(M_i M_j) d(\partial\Omega_i) + \sum_{i=1}^N T_i \left( \int_{\partial\Omega_i} \frac{\partial \ln(M_i M_j)}{\partial n} d(\partial\Omega_i) - \pi \delta_{ij} \right) = 0 \quad (4.12)$$

pour une géométrie dont la frontière comprend N éléments  $\partial\Omega_i$ , i variant de 1 à N et

où  $M_j$  est un point source (associé à l'inconnue  $T_j$ ) appartenant à la frontière,  
 $M_i$  est un point courant sur le morceau  $\partial\Omega_i$ ,  
 $\delta_{ij}$  est le symbole de Kronecker qui vaut 1 si  $i=j$  et 0 sinon,  
 $d(\partial\Omega_i)$  est l'élément différentiel de  $\partial\Omega_i$ .

Par ailleurs, pour un point  $M_k$  n'appartenant pas à la frontière et appartenant à l'intérieur du domaine résolu, domaine pouvant s'étendre à l'infini, comme on l'a déjà vu, on aura :

$$- 2\pi T_k = - \sum_{i=1}^N \frac{\partial T_i}{\partial n} \int_{\partial\Omega_i} \ln(M_i M_k) d(\partial\Omega_i) + \sum_{i=1}^N T_i \int_{\partial\Omega_i} \frac{\partial \ln(M_i M_k)}{\partial n} d(\partial\Omega_i) \quad (4.13)$$

L'équation (4.12) forme ce que l'on a appelé le "système de résolution" tandis que l'équation (4.13) forme le "système de Post-Processing". Quel que soit le système, nous avons deux intégrales matricielles (car elles vont être assemblées dans des matrices) à calculer, soit :

$$G_{li} = \int_{\partial\Omega_i} \ln(M_i M_l) d(\partial\Omega_i) \quad (4.14)$$

$$H_{li} = \int_{\partial\Omega_i} \frac{\partial \ln(M_i M_l)}{\partial n} d(\partial\Omega_i) \quad (4.15)$$

où :  $M_l$  est un point appartenant à la frontière ou non.

Une fois que notre formulation est ainsi décomposée, nous pouvons entreprendre sa description à l'aide du Générateur de Formulations. Comme on l'a vu au précédent chapitre, on distingue quatre grandes parties :

- La description de l'inconnue et des propriétés.
- La description des discrétisations autorisées.
- La description des intégrales.
- La description des systèmes matriciels.

Nous allons détailler ces parties pour la formulation qui nous intéresse. Pour chacune des parties, une figure montrera la partie concernée du fichier formulation tel qu'il est généré.

#### Première partie : description de l'inconnue et des propriétés

Il s'agit simplement pour le concepteur de décrire les composantes de sa formulation et de les regrouper dans des vecteurs qui caractériseront les blocs des systèmes matriciels. Pour notre formulation, il y a deux vecteurs  $I$  et  $\frac{\partial I}{\partial n}$ , chacun comportant une composante, respectivement  $T$  et  $\frac{\partial T}{\partial n}$ . Pour chaque composante, des informations de continuité et de composante liée ( $T$  est liée à  $\frac{\partial T}{\partial n}$  et vice-versa) sont données par le concepteur. De plus, pour cette formulation, il n'y a pas de propriétés, la [figure 4.13](#) montre la zone du fichier concernant cette première partie.

#### *Description de l'inconnue et des propriétés*

```

EQUA3PP : Laplacien sans source ordre 0 avec pp.
DONNEES GLOBALES INCONNUE
  2 NOMBRE DE COMPOSANTES
  2 NOMBRE DE VECTEURS
  1 NOMBRE DE VARIATIONS TEMPORELLES
  1 NOMBRE DE VARIATIONS D'ESPACE
LISTE INCONNUE
TK : Temperature en Kelvin
  1 2 1 Numéro, numéro de la comp. liée, indicateur de continuité.
DTKDN : Derivee normale de la temperature en Kelvin
  2 1 2
LISTE VECTEUR
I : Vecteur inconnu
  1 1 Nombre de composantes, pointeur sur liste des composantes.
DIDN : Vecteur derive
  1 2
DONNEES GLOBALES PROPRIETES
  0 NOMBRE DE PROPRIETES
  
```

**Figure 4.13**

#### Deuxième partie : description des discrétisations autorisées

Comme nous l'avons vu plus haut, la discrétisation numérique de cette formulation est formée par des segments constants (correspondant à l'opérateur non-programmable SEG CST dans notre Base de Données) et la discrétisation géométrique est formée par des segments linéaires (correspondant à SEGLIN).

Cette partie consiste à choisir, pour la formulation décrite et parmi les opérateurs de la Base de Données, quelles sont les familles topologiques autorisées, et pour chaque famille, quels sont les éléments de Référence autorisés. En plus des deux éléments de référence cités ci-dessus, au moins un élément de référence surfacique doit être autorisé si l'on veut se baser sur

un maillage intérieur pour le système de Post-Processing. La figure 4.14 représente le fichier formulation issu de cette partie.

#### *Description des discrétisations autorisées*

DONNEES GLOBALES DISCRETISATION

2 DIMENSION DE L'ESPACE

0 NB.DE FAMILLES PONCTUELLES AUTORISEES

1 NB.DE FAMILLES LINEIQUES AUTORISEES

1 NB.DE FAMILLES SURFACIQUES AUTORISEES

0 NB.DE FAMILLES VOLUMIQUES AUTORISEES

0 NB.DE FAMILLES MACRO-ELEMENTAIRES AUTORISEES

LISTE DISCRETISATION

ELEMENTS LINEIQUES

SEGMENT

1 0 2 *Numéro d'ordre de la famille, indicateur elt. sing., Nb. d'elts. de Réf.*

SEGLIN : Segment [-1;+1] (2 noeuds extrémités)

SEGCST : Segment [-1;+1] a un noeud central

ELEMENTS SURFACIQUES

QUADRILATERE

2 0 1

QUAQU8 : Quadrilatere a huit noeuds [+1;-1]

Figure 4.14

#### Troisième partie : description des intégrales

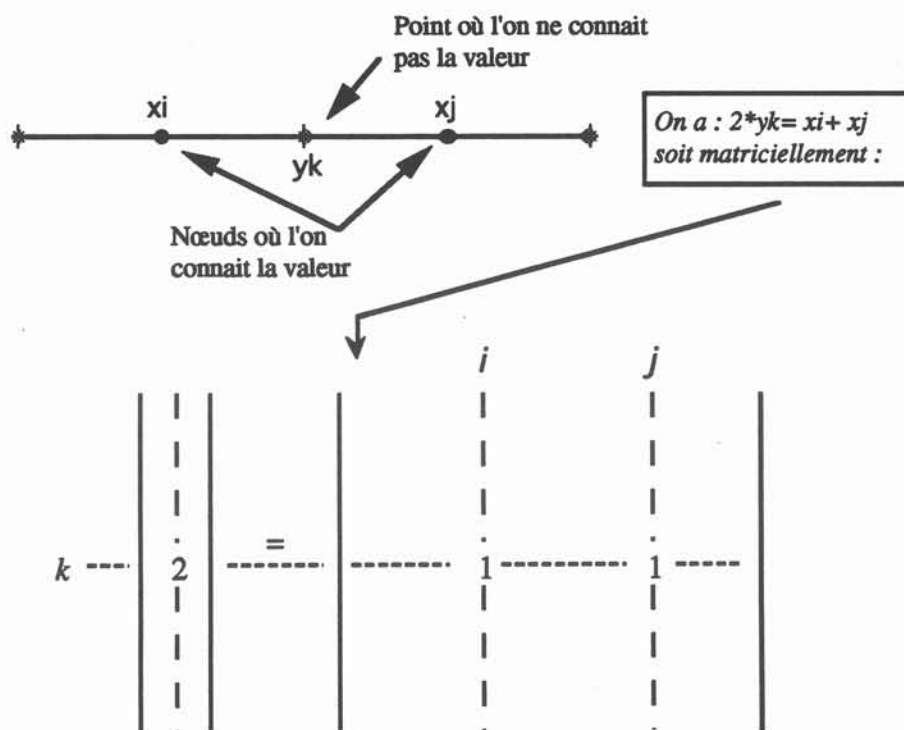
Nous rappelons que le principe général de description d'une intégrale est le suivant : à chaque intégrale est associé un opérateur critère de traitement qui détermine si l'intégrale doit être calculée selon l'équation et l'élément en cours. De plus, si elle doit être calculée, l'opérateur est chargé de renvoyer un numéro de pilote de traitement. Pour chaque traitement possible de l'intégrale, des opérateurs doivent être choisis dans la Base de Données. On spécifie éventuellement l'élément numérique sur lequel l'intégrale doit être calculée et l'élément géométrique sur lequel un traitement spécifique doit être effectué.

Nous avons vu lors de la décomposition de la formulation que nous avons principalement deux intégrales à calculer. Nous avons fait figurer en annexe A1 le calcul analytique de ces intégrales suivant que le point  $M_1$  appartienne (traitement singulier) ou n'appartienne pas (traitement régulier) à l'élément sur lequel on intègre. Pour chacune de ces intégrales matricielles, il y a deux traitements possibles suivant que l'équation (lieu du pôle de la fonction de Green) courante appartienne ou pas à l'élément en cours d'intégration. Dans notre Base de Données, l'opérateur de critère de traitement correspondant à ce critère d'appartenance est l'opérateur CRAPPA. Les traitements possibles sont tout deux analytiques mais correspondent à des opérateurs intégrants différents.



En plus de ces deux intégrales matricielles, nous devons décrire la manière dont on assemble le coefficient de collocation qui est, dans cette formulation, toujours égal à  $\pi$  pour les équations appartenant à la frontière et à  $2\pi$  pour les équations intérieures au domaine. L'opérateur critère de traitement sera différent suivant que le coefficient est destiné à être assemblé dans une matrice du système de résolution (CRCOLO) ou dans un vecteur du système de Post-Processing (CRPOST). Par contre le pilote de traitement et l'opérateur intégrant est le même dans les deux cas.

Il ne nous reste plus qu'à spécifier la manière dont seront calculées les inconnues aux points de la frontière qui ne font pas partie de la discrétisation numérique. On pourrait tout à fait utiliser la même formule pour les points intérieurs (avec toutefois un calcul singulier quelque peu différent) mais il est intéressant de montrer qu'il est possible d'utiliser une autre méthode grâce au Générateur. En effet, on peut considérer que la valeur de l'inconnue pour un point extrémité d'un élément frontière est la moyenne des valeurs de l'inconnue aux nœuds milieux des deux éléments auxquels il appartient. Ainsi, on aura besoin de deux "intégrales" pour ce Post-Processing des points frontières : l'une est chargée d'assembler 1 dans le vecteur multiplicatif des équations de Post-Processing et l'autre est chargée d'assembler 1 dans la matrice qui multiplie l'inconnue qui va servir à effectuer la moyenne, et ceci à chaque fois que l'équation courante appartient à l'élément courant. Le schéma de la [figure 4.15](#) illustre notre propos en ce qui concerne cette technique de lissage.

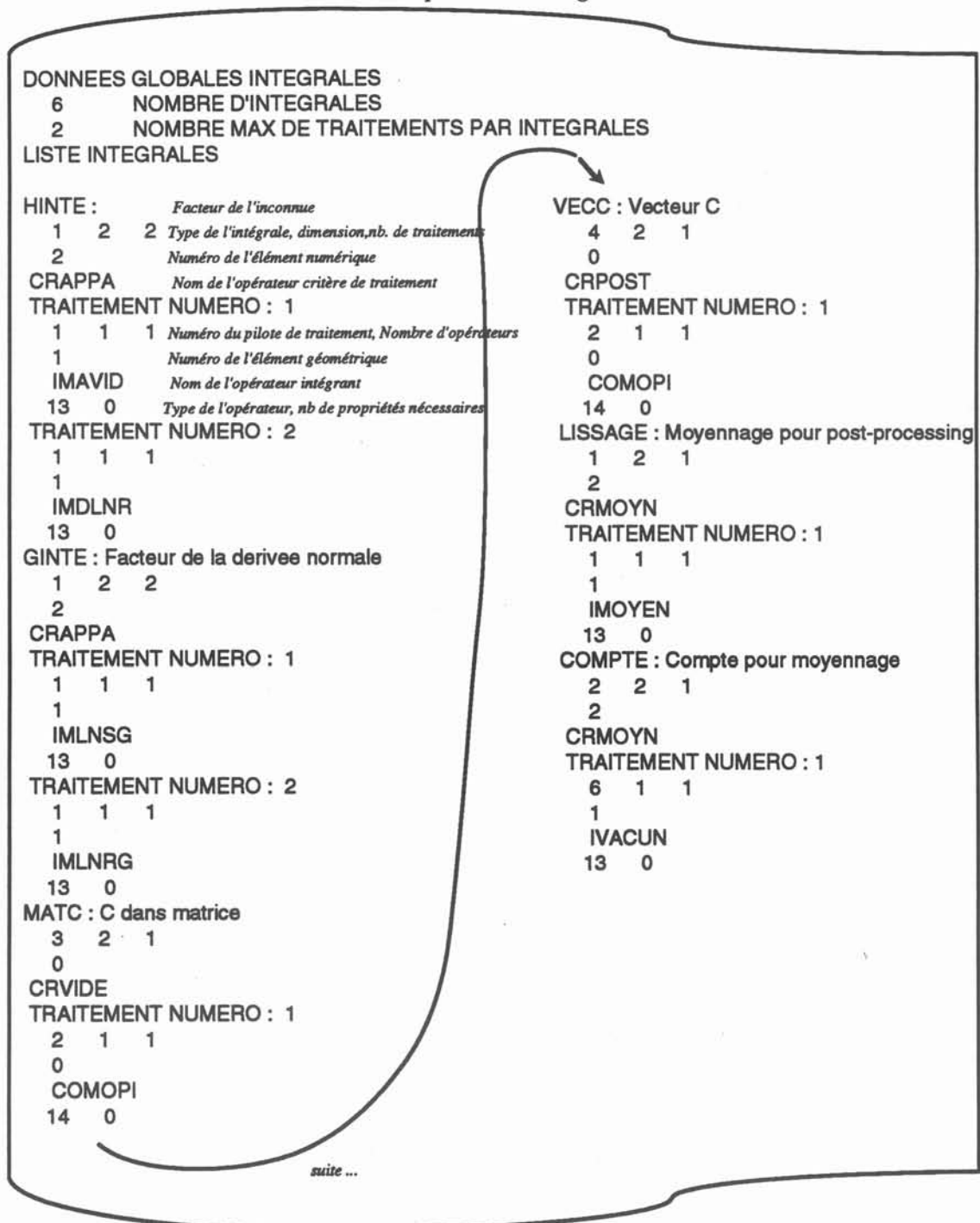


Calcul d'une moyenne à l'aide d'un assemblage matrice/vecteur

Figure 4.15

Quant à la figure 4.16, elle est le résultat de cette partie de description des intégrales dans le fichier issu du Générateur.

### Description des intégrales



**Figure 4.16**

#### Quatrième partie : description des système

Il s'agit de décrire les deux systèmes utilisés pour cette formulation, le premier correspondant à la résolution IF proprement dite (équation (4.12)), le second correspondant au Post-Processing des points intérieurs et frontières (équation (4.13) et lissage). Deux opérateurs sont présents dans la base de données et correspondent à ces deux systèmes classiques, c'est à dire :

une formulation matricielle du type :  $[H]\{I\} + [G]\{\partial I/\partial n\} = [K]$  pour le système de résolution (CSCLAS)

et une formulation matricielle du type :  $[C]\{I_{pp}\} = [H]\{I\} + [G]\{\partial I/\partial n\} + [K]$  pour le système de Post-Processing (CSPOST).

Pour chacun des deux systèmes, il s'agit de choisir dans la Base de Données l'opérateur de création des équations adéquats ainsi que les vecteurs de référence et annexes qui vont déterminer le nombre de blocs horizontaux, dans la vision "bloc" de notre description formelle, des matrices et des vecteurs. Le nombre de blocs étant fixé, le concepteur n'a plus alors qu'à répondre aux questions qui lui sont séquentiellement posées pour chacun des blocs, c'est à dire le nombre d'intégrales à assembler, et pour chacune des intégrales le nom d'un modèle multiplicatif.

La figure 4.17 est la partie du fichier correspondant à cette dernière partie de la génération de notre formulation. Comme on peut le constater sur cette figure, des intégrales peuvent être utilisées dans les deux systèmes, avec le même ou avec un coefficient multiplicatif différent, un même opérateur peut être utilisé dans deux intégrales différentes, les deux systèmes peuvent avoir des nombres de blocs différents, etc... La description matricielle des systèmes associée à la décomposition en opérateur des intégrales permet d'avoir une souplesse d'utilisation appréciable.

Les figures 4.13 à 4.17 (sauf 4.15) constituent le fichier formulation tel qu'il est lu par le solveur *Mif*. Ainsi, associé à une discrétisation compatible avec la formulation, c'est à dire ne comportant pas d'éléments de référence non autorisés par celle-ci, *Mif* donnera la solution de l'équation en tous points.

## Description matricielle des systèmes

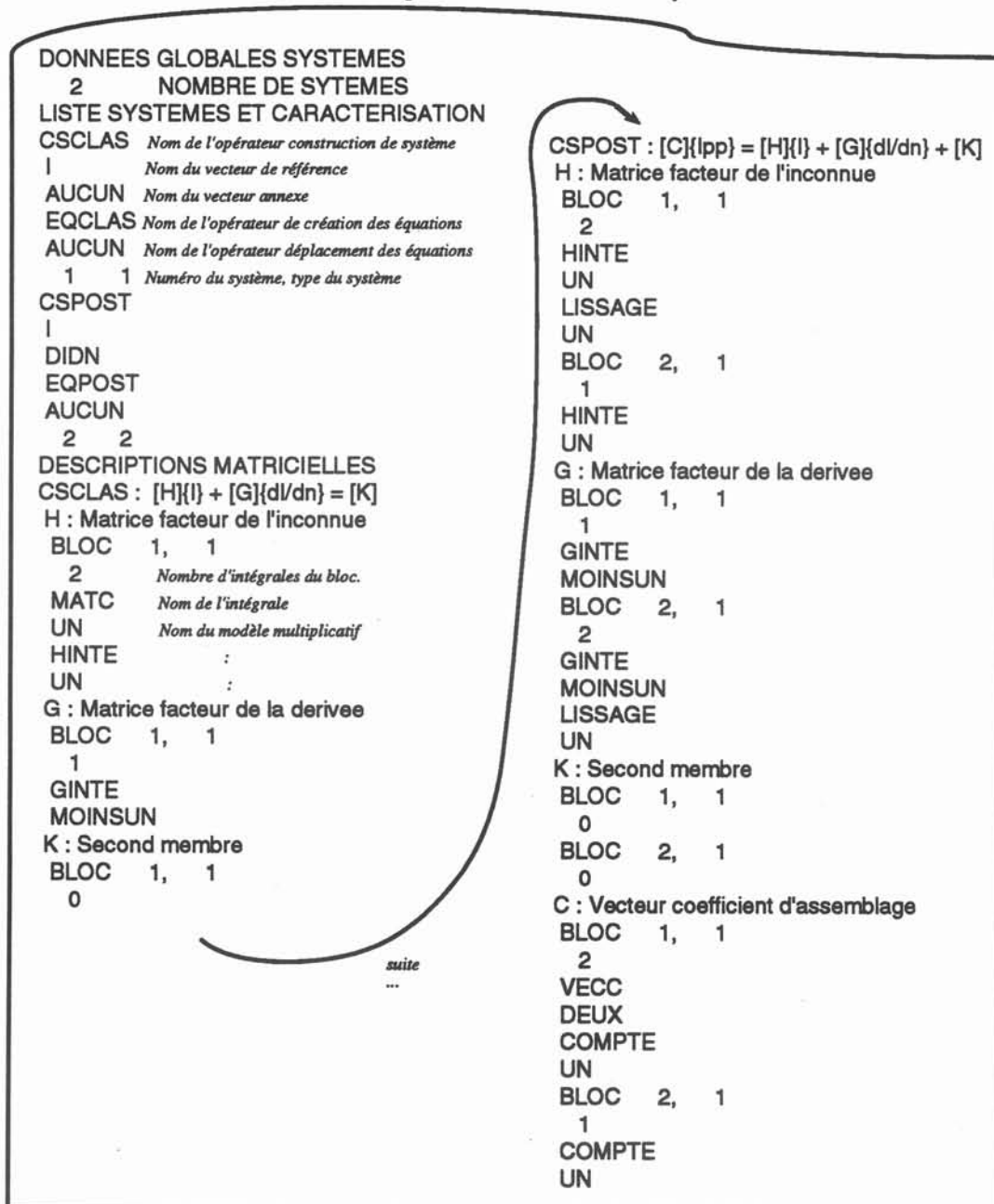
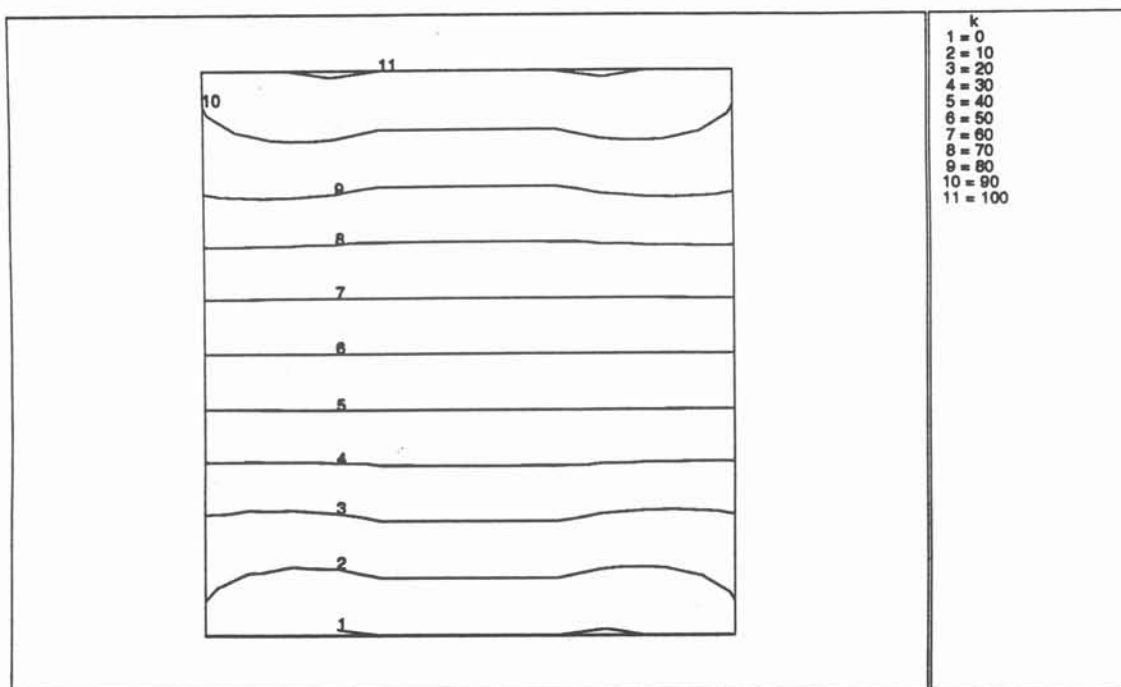


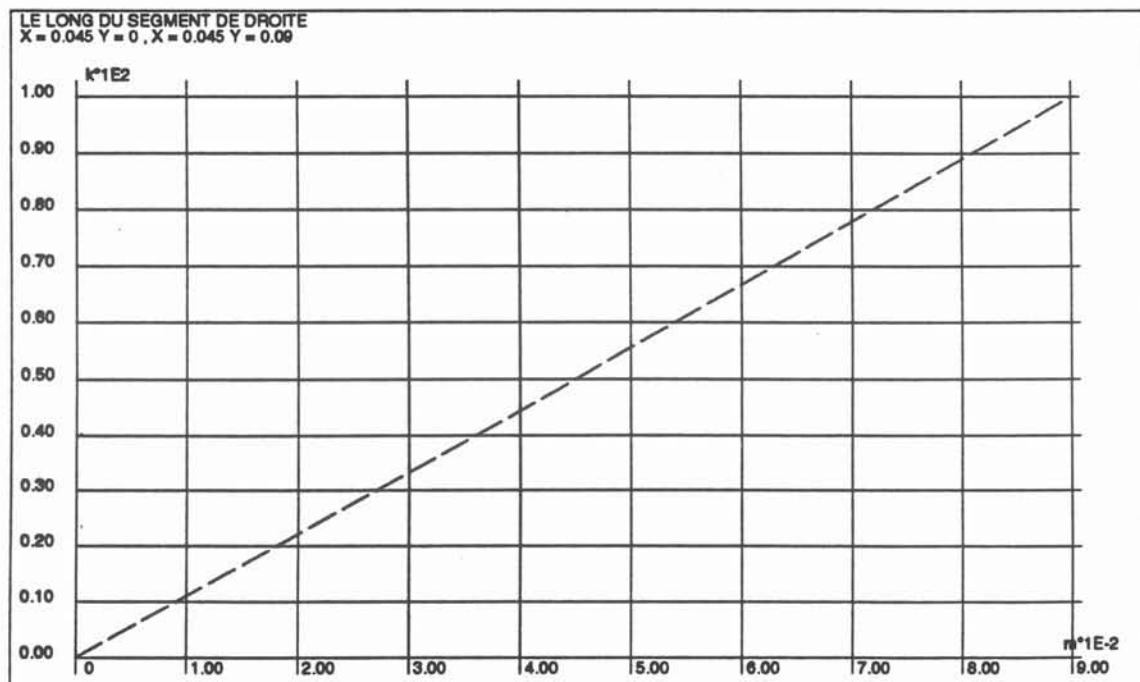
Figure 4.17

La figure 4.18 donne le résultat d'une telle résolution sur un carré dont les conditions aux limites sont 0 et 100 pour T, respectivement en bas et en haut, et 0 pour  $\frac{\partial T}{\partial n}$  sur les cotés. La solution d'un tel problème est bien connue, on doit obtenir des isovaleurs de T perpendiculaires aux cotés de flux nul et équiréparties entre 0 et 100. Comme on peut le constater sur la figure, c'est bien ce que l'on obtient, avec une petite imperfection sur les côtés, due au fait que les valeurs aux coins sont obtenues par moyenne des valeurs des deux éléments y concourant. L'erreur est donc uniquement due à la grossièreté de la discrétisation numérique et de plus, elle

est localisée uniquement aux coins comme on peut le constater sur la figure 4.19 qui montre la valeur de T le long du chemin verticale au milieu du carré. On constate que la courbe est bien parfaitement linéaire entre 0 et 100. Pour cette résolution, nous avons utilisé 12 segments répartis autour du carré (3 par coté). L'intérieur est maillé par 9 mailles carrées pour pouvoir utiliser le Post-Processeur de Flux-Expert.



**Figure 4.18**



**Figure 4.19**

## Deuxième étape : Laplace ordre 1 sans interfaces

Nous voulons dans cette étape autoriser des discrétisations numériques à l'ordre 1 et supprimer ainsi le problème du coin de l'étape précédente. Dans ce cas, les écritures des intégrales (4.14) et (4.15) de la première étape ne sont plus valables puisque les inconnues ( $T$  ou  $\frac{\partial T}{\partial n}$ ) ne peuvent plus être "sorties" des expressions. En effet, si la discrétisation numérique est linéaire, une inconnue  $I$  s'écrira sur un segment :

$$I = \alpha_1 I_1 + \alpha_2 I_2 \quad (4.16)$$

où :  $\alpha_1$  et  $\alpha_2$  sont les deux polynômes d'interpolation caractéristiques d'un segment linéaire,

$I_1$  et  $I_2$  sont les valeurs de l'inconnue  $I$  aux deux nœuds du segment.

Ainsi, les deux intégrales (4.14) et (4.15) ne donnent plus à assembler qu'un seul coefficient matriciel mais deux coefficients matriciels pour chaque couple équation-élément. Ces coefficients matriciels auront pour expression :

$$G_{li}^k = \int_{\partial\Omega_i} \ln(M_i M_l) \alpha_k d(\partial\Omega_i) \quad (4.17)$$

$$H_{li}^k = \int_{\partial\Omega_i} \frac{\partial \ln(M_i M_l)}{\partial n} \alpha_k d(\partial\Omega_i) \quad (4.18)$$

où :  $i$  est l'indice correspondant à l'élément,  
 $l$  est l'indice correspondant à l'équation,  
 $k$  est l'indice correspondant au nœud de l'élément.

Tout comme précédemment, nous avons deux cas différents pour chaque intégrale selon que l'équation appartienne à l'élément ou pas. Pour les cas singuliers (l'équation appartient à l'élément), le traitement envisagé est une nouvelle fois analytique et l'on trouvera en annexe A2 le détail du calcul de ces intégrales. Quant aux traitements réguliers, une technique numérique classique d'association points/poids d'intégration sera utilisée. Nous ne décrirons pas ici le fonctionnement interne du pilote de traitement numérique mais pour le concepteur, la description d'un traitement numérique n'est bien sûr pas identique à la description d'un traitement analytique où seul un opérateur intégrant doit être choisi. En effet, quatre opérateurs doivent être choisis dans la Base de Données pour définir un traitement numérique :

- l'opérateur "possibilité de répartition de points d'intégration" qui, comme son nom l'indique sert à déterminer le nombre, la répartition des points d'intégration et la valeur des poids d'intégration qui y sont relatifs. Dans l'exemple traité, l'opérateur SEGAU5 a été choisi



pour les deux traitements numériques, il permet d'utiliser 5 points d'intégration de Gauss-Legendre sur un segment.

- l'opérateur "argument de noyaux" qui est chargé de renvoyer la valeur de l'argument pour tous les points de calcul (id. les points d'intégration pour un traitement numérique), ici les composantes du vecteur reliant l'équation au point de calcul, et ceci pour les deux intégrales (opérateur ARRY2D)

- l'opérateur "noyau singulier" qui calcule la valeur des noyaux, c'est à dire  $\ln(M_i M_j)$  dans un cas (NYLOGV) et  $\frac{\partial \ln(M_i M_j)}{\partial n}$  dans l'autre (NYVNV2).

- enfin, l'opérateur intégrant qui effectue, pour tous les points de calculs, la multiplication du polynôme d'interpolation par le noyau et le poids d'intégration (IMNNOY) , le pilote numérique se chargeant d'affecter ces poids d'intégration par le Jacobien de la transformation géométrique nécessaire à l'intégration numérique.

#### *Une partie de la description des intégrales ...*

HINTE : Facteur de l'inconnue	GINTE : Facteur de la dérivée
1 2 2	1 2 2
2	2
CRAPPA	CRAPPA
TRAITEMENT NUMERO : 1	TRAITEMENT NUMERO : 1
1 1 1	1 1 1
1	1
IMAVID	IMLNLS
13 0	13 0
TRAITEMENT NUMERO : 2	TRAITEMENT NUMERO : 2
3 4 3	3 4 3
1	1
IMNNOY	IMNNOY
13 0	13 0
NYVNV2	NYLOGV
10 0	10 0
ARRY2D	ARRY2D
9 0	9 0
SEGAU5	SEGAU5
4 0	4 0
	SNGTK : Relation singulière Tk
	3 2 1
	0
	CRSING
	TRAITEMENT NUMERO : 1
	4 1 1
	0
	SGEGAL
	15 0

**Figure 4.20**

La figure 4.20 montre la description de ces deux intégrales ainsi qu'une troisième (SNGTK) qui n'apparaissait pas lors de la première étape. Cette dernière est chargée d'assurer la continuité de T au point où il existe une singularité. En effet, puisque les nœuds sont aux extrémités des éléments, il peut exister des discontinuités de valeurs limites, ce qui, comme on l'a vu au chapitre 1, crée des singularités. Cela implique deux changements par rapport à la formulation précédente. D'une part, les attributs de création et de déplacement des équations du système de résolution (qui demeure inchangé) doivent tenir compte de l'éventuelle présence de singularité dans la discrétisation, cela implique un nouvel opérateur de création des équations (EQCOIN) et un opérateur de déplacement des équations (DPSGLN). D'autre part, la description matricielle du système de résolution s'enrichit de l'intégrale singulière SNGTK. La figure 4.21 montre la partie du fichier formulation concernée par ces modifications par rapport à ces modifications, la description de CSPOST étant inchangée.

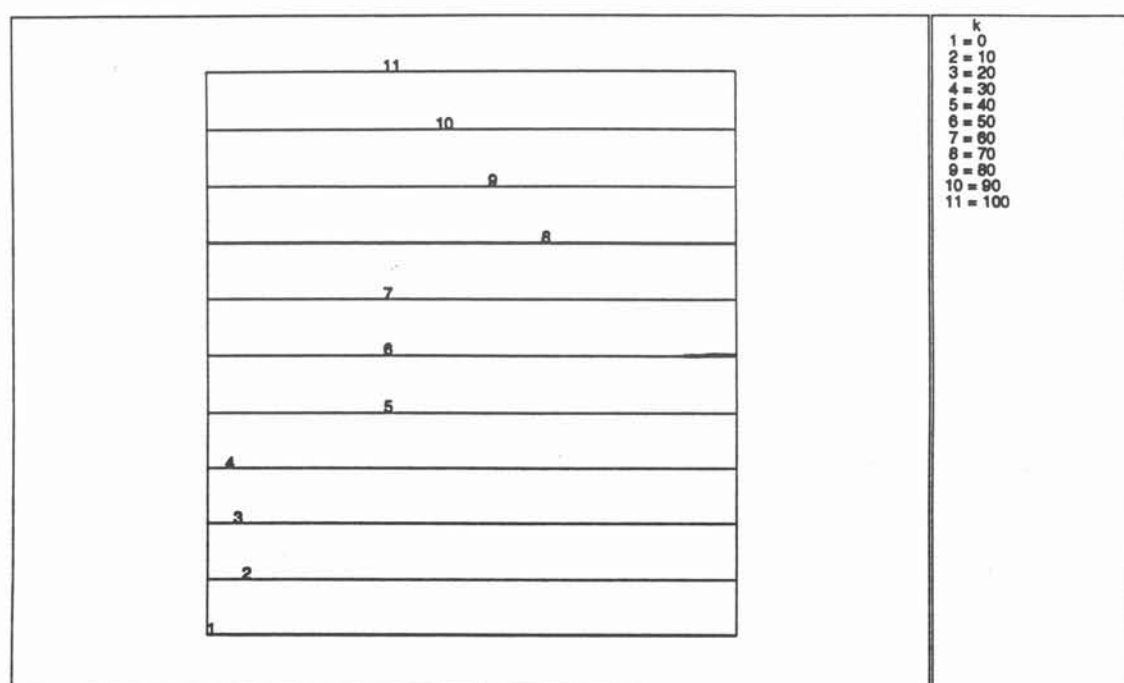
*Une partie de la description des systèmes ...*

DONNEES GLOBALES	
2	NOMBRE DE SYTEMES
LISTE SYSTEMES	DESCRIPTIONS MATRICIELLES
CSCLAS	CSCLAS : $[H]\{I\} + [G]\{dl/dn\} = [K]$
1	H : Matrice facteur de l'inconnue
AUCUN	BLOC 1, 1
EQCOIN	3
DPSGLN	MATC
1 1	UN
CSPOST	HINTE
1	UN
DIDN	SNGTK
EQPOST	UN
AUCUN	G : Matrice facteur de la derivee
1 2	BLOC 1, 1
	1
	GINTE
	MOINSUN
	K : Second membre
	BLOC 1, 1
	0

Figure 4.21

En plus des modifications décrites ci-dessus, il faut signaler que, puisque les équations peuvent être positionnées dans des angles (car elles n'appartiennent plus obligatoirement aux centres des segments), les coefficients de collocation ne sont plus constants, égaux à  $\pi$ . Un autre opérateur de collocation est donc utilisé pour tenir compte de cette éventualité, il calcule l'angle que fait la frontière à l'endroit où est positionnée l'équation, puisque, nous l'avons vu au chapitre 1, le coefficient est égal à cet angle dans le cas de l'équation de Laplace.

La géométrie sur laquelle nous allons tester la formulation ainsi décrite est la même que celle utilisée pour la précédente étape (un carré). Les conditions aux limites étant appliquées sur chacun des côtés de ce carré, les coins sont susceptibles de donner lieu à des singularités, et l'on utilise donc une discrétisation disposant de double-nœuds aux coins du carré comme le montre la [figure 1.3](#) du premier chapitre. La [figure 4.22](#) montre bien que, contrairement au traitement de la première étape, les valeurs aux coins sont parfaitement calculées, le double-nœud permettant de calculer exactement les deux valeurs de  $\frac{\partial T}{\partial n}$  de part et d'autre des coins.



**Figure 4.22**

### Troisième étape : Darcy ordre 1 sans interfaces

Dans les deux précédentes étapes, la conductibilité thermique du matériau modélisé n'apparaissait pas car elle était supposée constante. Nous allons considérer maintenant une conductibilité thermique orthotrope, c'est à dire que les directions d'anisotropie sont parallèles aux axes de la géométrie,  $k$  est donc un tenseur diagonal que l'on peut exprimer ainsi :

$$k = \begin{bmatrix} k_x & 0 \\ 0 & k_y \end{bmatrix}.$$

L'équation de départ (4.10) devient alors l'équation de Darcy, c'est à dire :

$$k_x \frac{\partial^2 T}{\partial x^2} + k_y \frac{\partial^2 T}{\partial y^2} = 0 \quad (4.19)$$

dont nous avons vu au premier chapitre qu'elle admettait pour fonction de Green :

$$G(r) = \frac{1}{\sqrt{k_x k_y}} \frac{1}{2\pi} \text{Ln}\left(\frac{1}{r}\right) \quad (4.20)$$

où  $r = \sqrt{\frac{1}{k_x} r_x^2 + \frac{1}{k_y} r_y^2}$ ,

$r_x$  et  $r_y$  étant les composantes du vecteur reliant le pôle de la fonction de Green au point de calcul.

Avec le couple d'inconnues  $T$  et  $P$  ( $P = k_x \frac{\partial T}{\partial x} n_x + k_y \frac{\partial T}{\partial y} n_y$  où  $n_x$  et  $n_y$  sont les cosinus directeurs de la normale au point où l'on calcule) la formulation matricielle de la résolution de cette équation par la Méthode des Intégrales de Frontières est strictement identique à celle de l'équation de Laplace. De plus, on peut montrer que, dans le cas où les directions d'anisotropie sont parallèles aux frontières de la géométrie, le calcul du coefficient de collocation donne des résultats identiques au cas de l'équation de Laplace [10:BRUCH]. Si l'on se place dans ce dernier cas, on constate que la seule différence de cette étape avec la précédente se situe au niveau de la fonction de Green.

Ainsi, pour les deux traitements numériques, il suffit donc choisir dans la Base de Données l'opérateur "noyau singulier" adéquat, que l'on aura préalablement programmé s'il n'existait pas. Quant aux traitements analytiques des deux intégrales  $H$  et  $G$  quand elles deviennent singulières, ils sont tout à fait similaires à ceux effectués pour l'étape précédente. La [figure 4.23](#) montre ainsi la zone du fichier formulation changée par rapport au précédent, tout le reste demeurant identique.

Ce qui change entre l'équation de Laplace et l'équation de Darcy ...

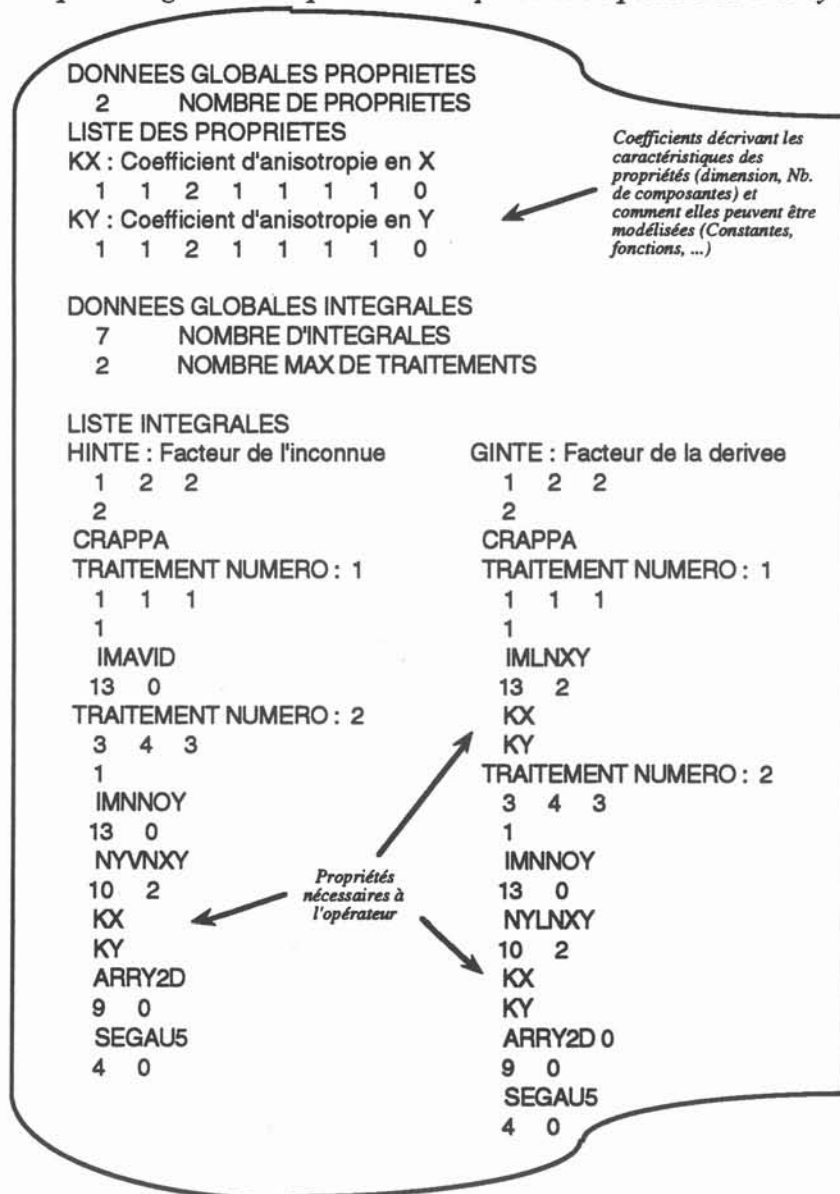
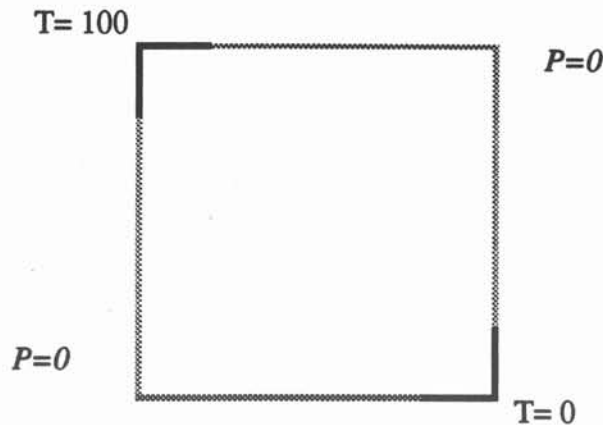


Figure 4.23

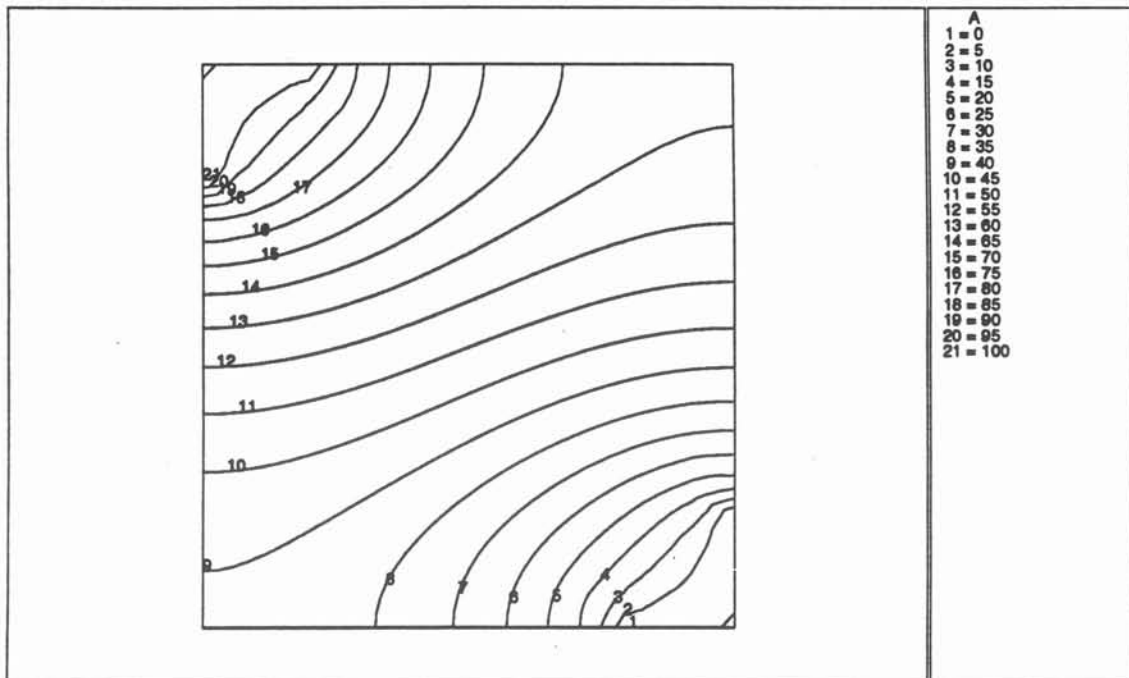
Pour tester cette formulation, qui donne les mêmes résultats que la précédente dans le cas où  $k_x=k_y$ , nous allons confronter les résultats qu'elles donnent avec ceux que donnent Flux-Expert sur un cas simple. On considère le problème dont les conditions aux limites sont schématisées par la [figure 4.24](#), avec  $k_x=200$  et  $k_y = 100$ .

### Conditions aux limites pour le problème orthotropique



**Figure 4.24**

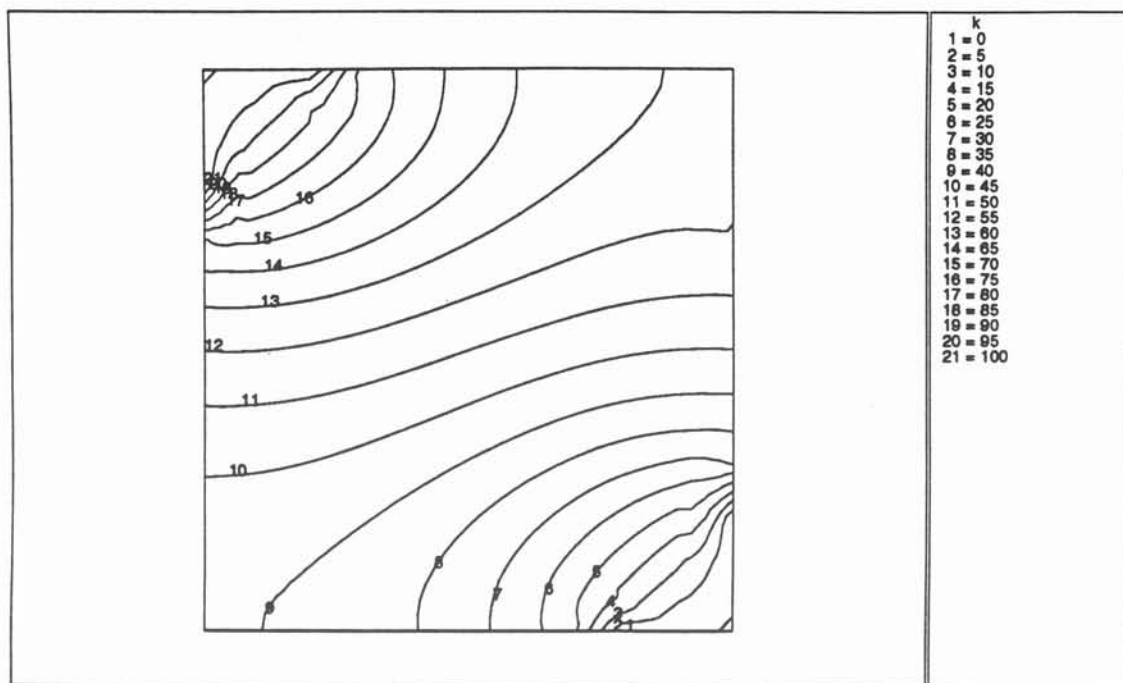
Les deux figures suivantes (4.25 et 4.26) montrent les isovaleurs de  $T$  pour ce problème résolu respectivement par Flux-Expert et Green-Expert. On y constate une bonne concordance malgré un léger décalage facilement observable sur les isovaleurs 9 et 13. Cette différence peut être expliquée par un traitement différent des points de rencontre entre les deux types de condition aux limites (CAL). En effet, Flux-Expert ne permet pas qu'un seul point ait une double CAL sur  $T$  et sa dérivée alors que Green-Expert le permet.



**Problème résolu par Flux-Expert**

**Figure 4.25**





**Problème résolu par Green-Expert**

**Figure 4.26**

## Dernière étape : Darcy ordre 1 avec interfaces

Dans les trois précédents exemples, la formulation était à chaque fois appliquée sur une unique frontière, le domaine modélisé possédant des propriétés physiques constantes permettant une résolution par la MIF. Dans le cas où le domaine à modéliser est décomposable en sous-domaines dans lesquels l'application de la formulation est possible, la résolution du problème sur la totalité du domaine est possible. Alors, la seule différence entre la formulation de l'étape précédente et celle que l'on veut générer ici est l'ajout d'intégrales gérant la continuité (ou la discontinuité) des composantes de l'inconnue à travers les interfaces séparant les sous-domaines.

*Rajout d'une intégrale pour la gestion des interfaces ...*

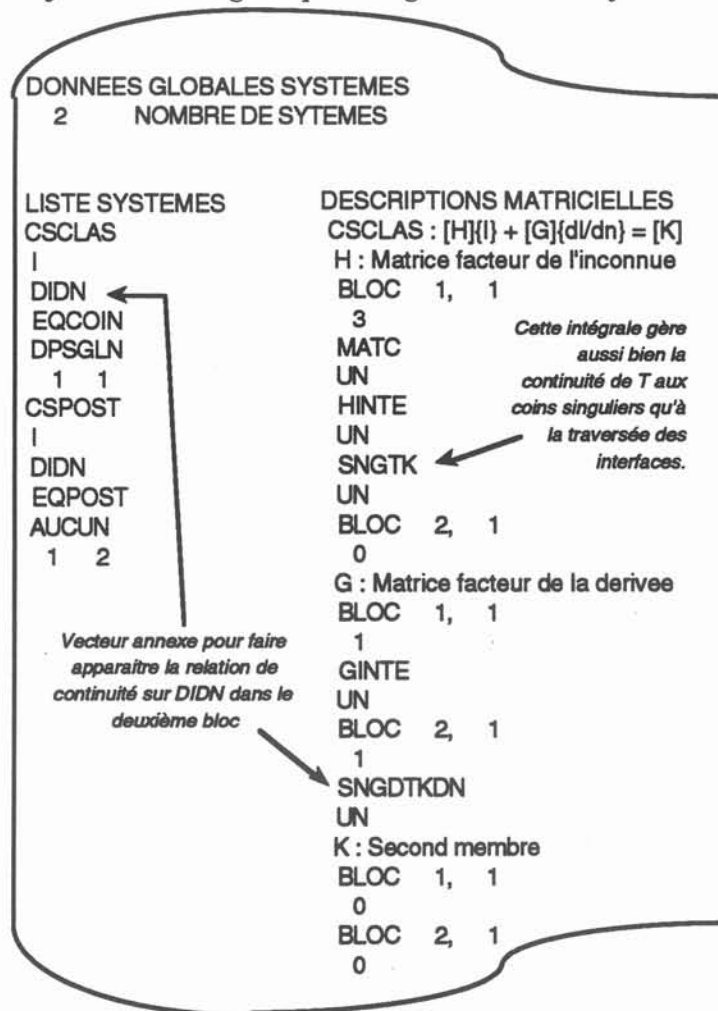
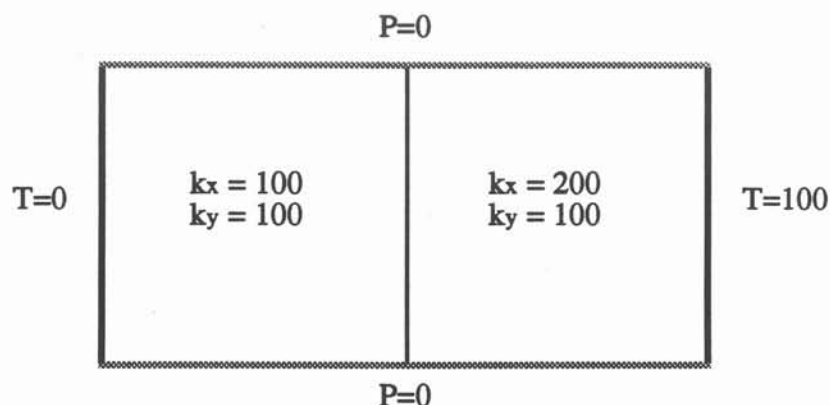


Figure 4.27

Les équations qui gèrent les discontinuités d'interfaces (comme d'ailleurs les singularités de coins ou d'arêtes) sont caractérisées dans l'opérateur de création des équations. En l'occurrence, nous avons conservé l'opérateur de l'étape précédente (EQCOIN) en élargissant, en quelque sorte, son domaine de compétence puisqu'il devient capable de gérer non seulement les singularités de coins mais aussi les singularités d'interfaces. C'est d'ailleurs la même intégrale qui gère, pour  $T$ , les deux types de singularités. Par contre, il nous faut rajouter l'intégrale qui gère la continuité de  $P$  à travers une interface (en fait, avec l'opposition des normales extérieures de deux sous-domaines accolés,  $S_1$  et  $S_2$ , on a  $P_1 + P_2 = 0$ ). Le rajout de cette intégrale dans la formulation matricielle nous oblige à considérer un deuxième bloc horizontal dans la description de CSCLAS, comme le montre la [figure 4.27](#).

Afin de tester cette formulation, on résout un problème dont on connaît la solution analytique. La définition du problème est en [figure 4.28](#).



[Figure 4.28](#)

La solution de ce problème est connue, puisqu'il est en fait 1D. Si l'on trace une droite horizontale le long de toute la géométrie, on doit trouver une ligne brisée entre 0 et 100, formée par deux segments. La brisure se situe à l'interface des deux domaines (ici au milieu) et la valeur de  $T$  à cet endroit est donnée par l'équation  $100 x = 200 (100 - x)$ , soit  $x = 66,6667$ .

C'est effectivement ce que la formulation décrite donne quand on l'applique au problème de la [figure 4.28](#) comme on peut le voir sur les [figure 4.29](#) et [4.30](#) qui montrent respectivement les isovaleurs de  $T$  et les valeurs de  $T$  le long d'un segment horizontal coupant la géométrie sur toute sa longueur.

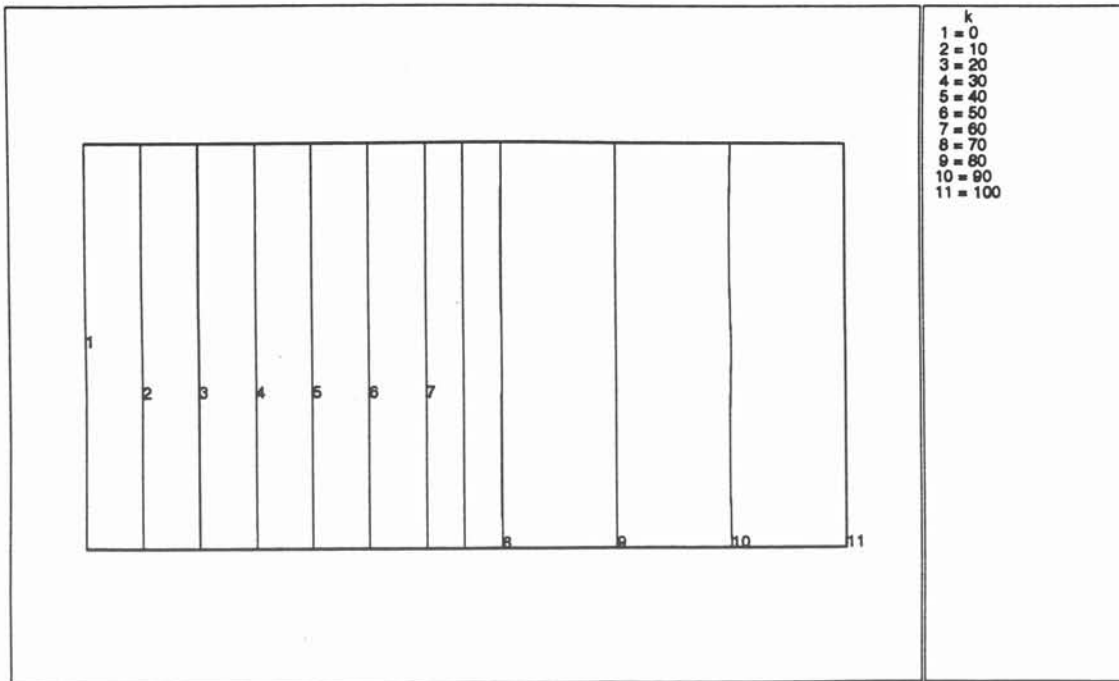


Figure 4.29

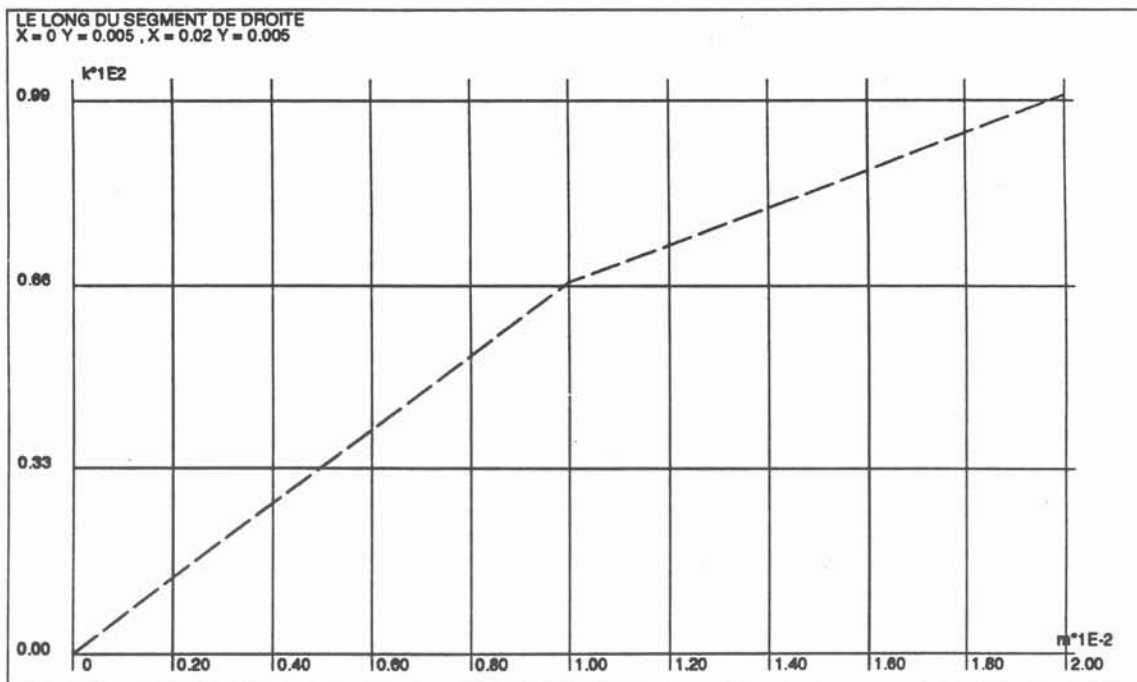
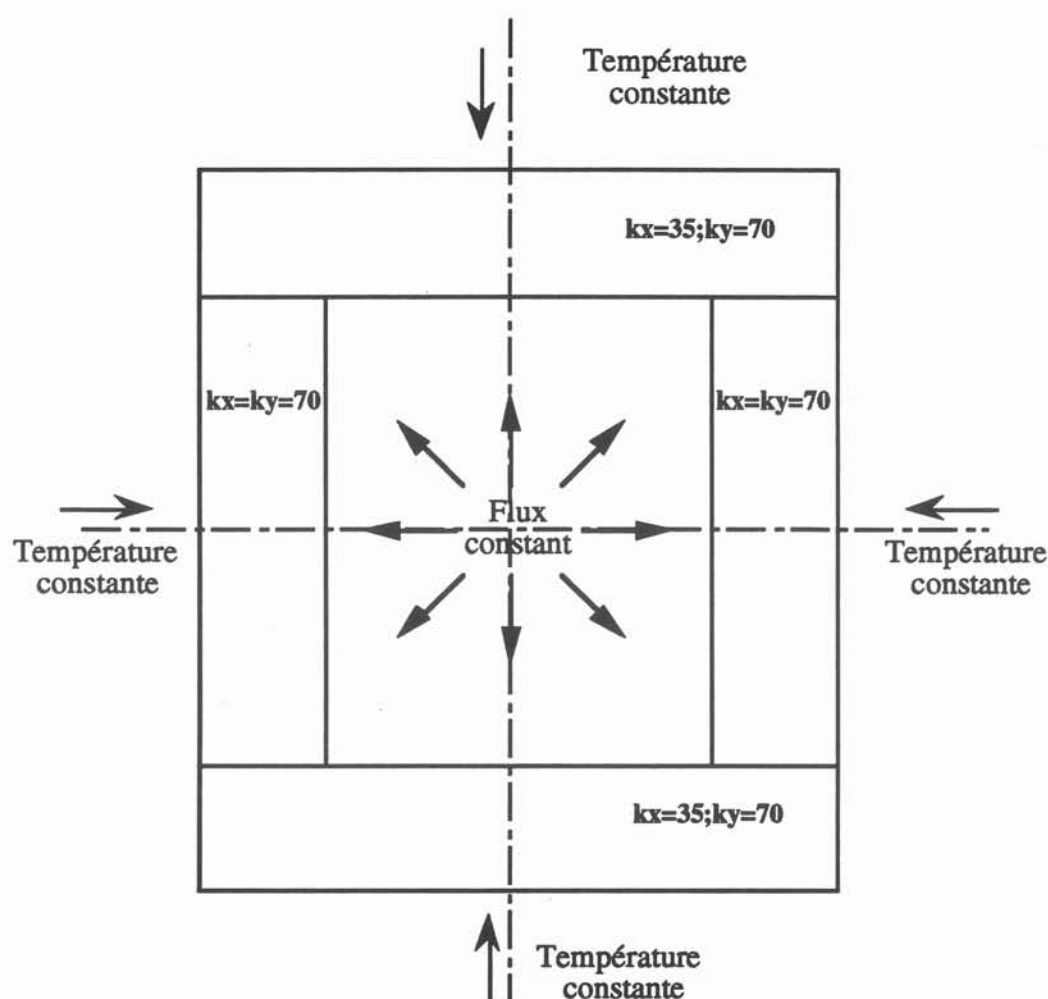


Figure 4.30

Afin de tester cette dernière formulation 2D sur un cas plus concret, nous allons résoudre à l'aide de Green-Expert et à l'aide de Flux-Expert le problème défini par la [figure 4.31](#).

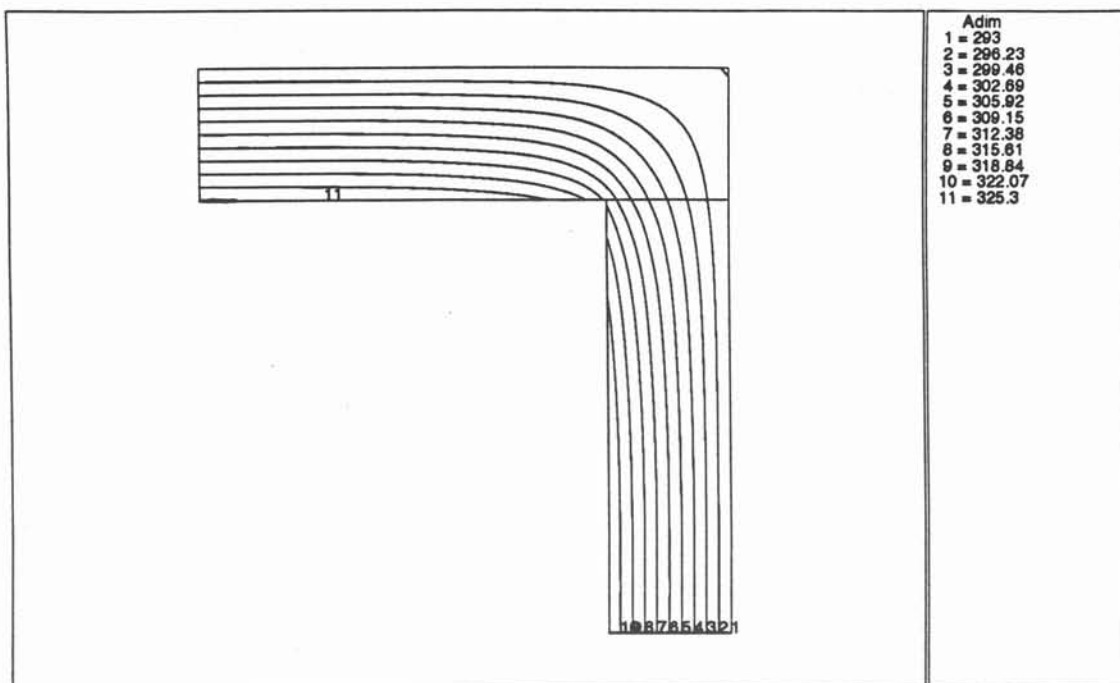


**Cas test pour l'équation de Darcy ordre 1 avec interfaces**

**Figure 4.31**

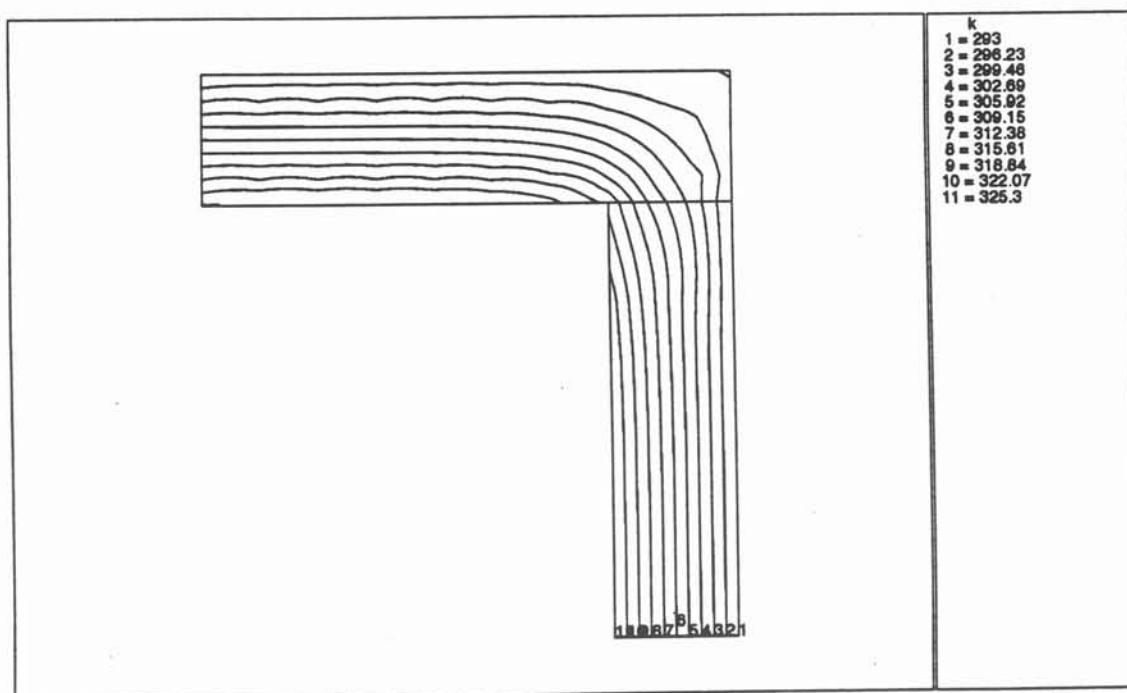
Les deux axes de symétries nous permettent de ne modéliser qu'un quart de la géométrie, en imposant un flux nul au niveau des axes.

La [figure 4.32](#) montre les isovaleurs de l'inconnue résolue par Flux-Expert et la [figure 4.33](#) celles résolues par Green-Expert. On observe une très bonne concordance des résultats compte tenu des différences entre les deux logiciels déjà signalées. D'une part, Flux-Expert résout avec des éléments numériques quadratiques alors que notre formulation utilise des éléments linéaires (ce qui pourrait expliquer les légères oscillations observées sur nos résultats). D'autre part, les double-nœuds présents aux coins de la géométrie dans la résolution de Green-Expert et inexistants dans Flux-Expert peuvent expliquer la très légère différence entre les valeurs maximum (325,339 pour Flux-Expert et 325,313 pour Green-Expert, soit 8 millièmes de pour-cent d'écart relatif).



Résolution par Flux-Expert

Figure 4.32



Résolution par Green-Expert

Figure 4.33



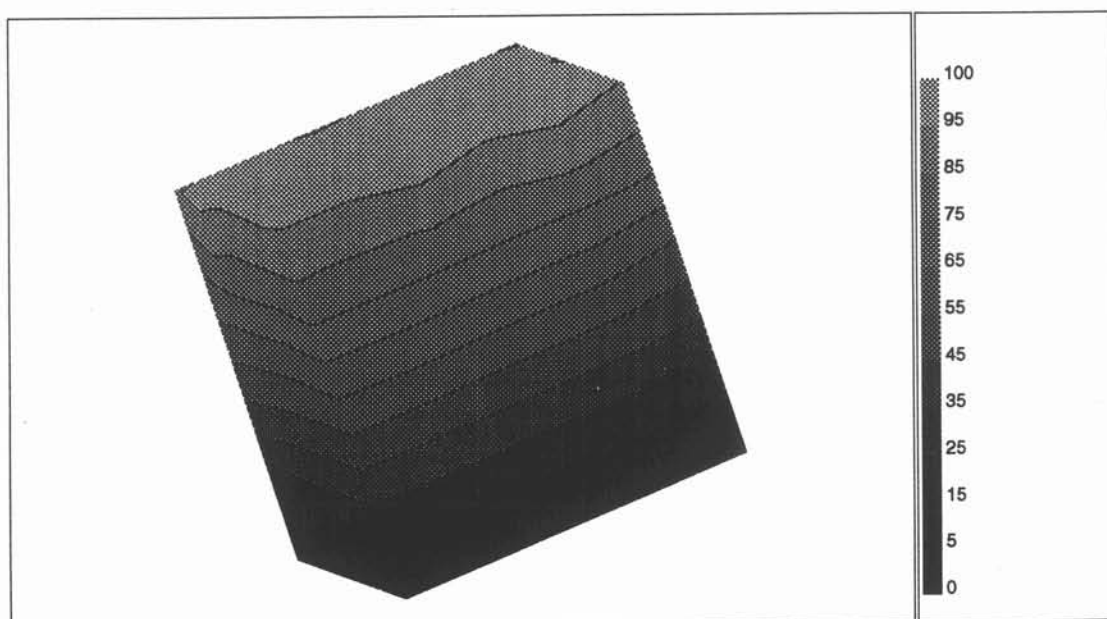
### 3.3 Un exemple de résolution 3D

La génération d'une formulation 3D ne pose pas plus de problème qu'une formulation 2D et notre solveur, complètement général, peut résoudre indifféremment toute formulation issue du Générateur. A titre d'exemple, la [figure 4.35](#) montre la totalité du fichier formulation de l'équation de Laplace à l'ordre 0, c'est à dire l'équivalent 3D de la première étape du paragraphe précédent. On peut constater que la différence essentielle de cette formulation avec celle du 2D réside dans la description des intégrales et plus particulièrement ce qui concerne la fonction de Green qui, pour l'équation de Laplace 3D est :

$$G_{M'}(M) = \frac{1}{4\pi M'M}$$

La fonction de Green changeant, on devra, non seulement changer les opérateurs la calculant dans les traitements numériques (Poids/points d'intégration, Argument, Noyau), mais aussi, avoir un opérateur calculant l'intégrale singulière de  $G_{M'}(M)$  sur un élément surfacique (ici, nous avons choisi le triangle). On pourra se reporter à l'annexe A3 pour consulter les calculs relatifs à cette intégrale singulière. Les éléments de référence sont bien sûr relatifs au 3D, c'est à dire que les segments ont laissé la place aux triangles. Par contre la description matricielle des système est tout à fait similaire à la formulation 2D.

La [figure 4.34](#) nous montre les isovaleurs de T issues d'une résolution de cette formulation sur un cube avec des CAL de flux nuls sur les 4 faces verticales et des CAL de Dirichlet entre 0 et 100 sur les faces horizontales. Les légères imperfections observées sont dues à la grossièreté de la discrétisation numérique (ordre 0) et le lissage que cela implique (comme en 2D) associé au fait que le maillage de frontière est composé de triangles irrégulièrement répartis sur les six faces.



**Figure 4.34**

## Fichier formulation complet de l'équation de Laplace 3D, ordre 0

```

EQUASPP : Thermique 3D ordre 0 + pp
DONNEES GLOBALES INCONNUE
1
2 INDICATEUR DESCRIPTION SIMPLIFIEE
2 NOMBRE DE COMPOSANTES
2 NOMBRE DE VECTEURS
1 NOMBRE DE VARIATIONS TEMPORELLES
1 NOMBRE DE VARIATIONS D'ESPACE
LISTE INCONNUE
TK : Temperature en Kelvin
1 2 1
DTKDN : Derivee normale de la temperature en Kelvin
2 1 2
LISTE VECTEUR
I : Vecteur inconnue
1 1
DIDN : Vecteur derivee
1 2
DONNEES GLOBALES DISCRETISATION
3
0 DIMENSION DE L'ESPACE
0 MODE DE DISCRETISATION EXPERT
0 INDICATEUR SOUS-DOMAINES AUTORISEES
0 INDICATEUR TYPE COLLOCATION
0 NB.DE FAMILLES PONCTUELLES AUTORISEES
0 NB.DE FAMILLES LINEAIQUES AUTORISEES
1 NB.DE FAMILLES SURFACIQUES AUTORISEES
1 NB.DE FAMILLES VOLUMIQUES AUTORISEES
0 NB.DE FAMILLES MACRO-ELEMENTAIRES AUTORISEES
LISTE DISCRETISATION
ELEMENTS SURFACIQUES
TRIANGLE
1 0 2
TRILIN : Triangle lineaire [0,+1,+1] a 3 noeuds
TRICST : Triangle constant [0,+1,+1] a un noeud central
ELEMENT VOLUMIQUES
TETRAEDRE
2 0 1
TETOC : Tetraedre quadratique Centre [0,+1,+1,+1] a 15 noeuds
DONNEES GLOBALES PROPRIETES
0 NOMBRE DE PROPRIETES
LISTE DES PROPRIETES
DONNEES GLOBALES INTEGRALES
0 NOMBRE D'INTEGRALES
2 NOMBRE MAX DE TRAITEMENTS
LISTE INTEGRALES
HINTE : Facteur de l'inconnue
1 3 2
2
CRAPPA
TRAITEMENT NUMERO : 1
1 1 1
1
IMAVID
13 0
TRAITEMENT NUMERO : 2
3 4 3
1
IMNNOY
13 0
NYVNV3
10 0
ARRY3D
9 0
TRGAU7
4 0
GINTE : Facteur de la derivee normale de l'inconnue
1 3 2
2
CRAPPA
TRAITEMENT NUMERO : 1
1 1 1
1
IM1RTG
13 0
TRAITEMENT NUMERO : 2
3 4 3
1
IMNNOY
13 0
NYVNVV
10 0
ARRY3D
9 0
TRGAU7
4 0
MATC : C dans la matrice
3 3 1
0
CROCOLO
TRAITEMENT NUMERO : 1
2 1 1
0
CODEPI
14 0
VECC : Vecteur C
4 3 1
0
CRPOST
TRAITEMENT NUMERO : 1
2 1 1
0
CODEPI
14 0
LISSAGE : Moyennage pour post-processing
1 3 1
2
CRMOYN
TRAITEMENT NUMERO : 1
1 1 1
1
IMOYEN
13 0
COMPTTE : Compte pour moyennage
2 3 1
2
CRMOYN
TRAITEMENT NUMERO : 1
6 1 1
1
IVACUN
13 0
DONNEES GLOBALES SYSTEMES
2 NOMBRE DE SYTEMES
LISTE SYSTEMES
CSCLAS
1
AUCUN
ECLAS
AUCUN
1 1
CSPOST
1
DIDN
ECPST
AUCUN
1 2
DESCRIPTIONS MATRICIELLES
CSCLAS : [H][I] + [G][d/dn] = [K]
H : Matrice facteur de l'inconnue
BLOC 1, 1
2
MATC
UN
HINTE
UN
G : Matrice facteur de la derivee
BLOC 1, 1
1
GINTE
UN
K : Second membre
BLOC 1, 1
0
CSPOST : [C][pp] = [H][I] + [G][d/dn] + [K]
H : Matrice facteur de l'inconnue
BLOC 1, 1
2
HINTE
UN
LISSAGE
UN
BLOC 2, 1
1
HINTE
UN
G : Matrice facteur de la derivee
BLOC 1, 1
1
GINTE
UN
BLOC 2, 1
2
GINTE
UN
LISSAGE
UN
K : Second membre
BLOC 1, 1
0
BLOC 2, 1
0
C : Vecteur coefficient d'assemblage
BLOC 1, 1
2
VECC
MOINSDEUX
COMPTTE
UN
BLOC 2, 1
1
COMPTTE
UN
===== EQUATION_IF TERMINEE

```

Figure 4.35

La simulation numérique est un outil qui peut s'insérer en amont de la chaîne de conception, comme aide à la compréhension d'un phénomène, ou en aval, comme aide à l'optimisation d'un procédé industriel. Il demeure encore des réticences dans les laboratoires et les entreprises à l'utilisation de cet outil. Ces réticences sont bien souvent fondées sur le manque d'adéquation du logiciel de simulation aux problèmes réels.

Nous avons voulu, avec l'élaboration de Green-Expert, offrir, aussi bien au concepteur d'une modélisation qu'à son utilisateur, un outil capable de s'adapter à ses propres besoins. Le choix de la Méthode des Intégrales de Frontières (MIF) s'inscrit aussi bien dans une dynamique de recherche que dans une logique industrielle. Cela devra permettre, à court terme, d'élargir le champ d'application des formulations résolubles en couplant notre solveur à celui qui était issu d'une première recherche axée sur la Méthode des Éléments Finis (MEF) : Flux-Expert.

Nous avons montré comment les deux méthodes numériques pouvaient avantageusement être couplées sur des problèmes où ni l'une ni l'autre ne seraient complètement adaptées. Nous avons vu aussi que, bien qu'elle soit moins systématique que la Méthode des Éléments Finis, la Méthode des Intégrales de Frontières pouvaient fort bien faire l'objet d'une analyse méthodologique permettant l'élaboration d'un outil, Green-Expert, basé sur un couple Générateur de Formulations/Solveur Généralisé. Cet outil n'a bien sûr pas encore la maturité d'un code industriel, un certain temps est nécessaire afin que les quelques quatre vingt mille lignes de Fortran qui le composent soient confrontées à la réalité d'une utilisation pratique et intense. Des formulations plus complexes, avec des éléments quadratiques, en 3D, doivent être développées, et nous pourrions alors envisager la suite de ce travail.

Il existe principalement deux axes de développement :

- Le multi-méthode : il s'agit de faire en sorte qu'un Solveur Généralisé soit capable de résoudre des problèmes dont la formulation fait appel à des méthodes numériques différentes. Notre expérience du couplage MEF-MIF nous laisse penser qu'une des difficultés principales reste le mélange des différentes discrétisations nécessaires à l'application des méthodes. En ce qui concerne le couplage MEF-MIF, le traitement des singularités pour la MIF complique la tâche et il est possible qu'une utilisation systématique d'éléments discontinus (Cf §1) soit la solution la plus viable.

• L'aide à la modélisation : il s'agit plutôt ici d'un travail concernant le (ou les) Générateur de Formulations. Notre objectif est de se rapprocher de la solution optimale dans l'espace adéquation-simplicité (Cf §1). D'une part, nous pouvons envisager une amélioration de l'interface homme-machine des outils existants de Génération des Formulations qui ne fonctionnent que dans un mode texte; cela concerne un secteur de l'informatique qui, bien qu'important, n'était pas dans nos axes de réflexion. D'autre part, il est clair que, plus les formulations envisageables à l'aide d'outils de Génération seront complexes, plus l'aide à la Génération de ces formulations sera indispensable. Cette aide ne pourra qu'être basée sur des principe d'intelligence artificielle car les compétences nécessaires à la modélisation numérique sont vastes et fortement disséminées dans les trois domaines qui forment le triptyque de toute modélisation : mathématiques, physique, numérique.

Le domaine de la simulation numérique des phénomènes physiques n'a sans doute pas autant bénéficié du formidable essor de l'industrie du logiciel que certain autres secteurs de l'informatique scientifique comme l'imagerie, la Conception Assistée par Ordinateur (CAO) ou même, récemment, la réalité virtuelle. L'exemple assez particulier du calcul des structures illustre assez bien comment devra évoluer la simulation numérique des autres phénomènes physiques. En effet, dans le domaine de la mécanique, il n'est pas rare de voir la simulation numérique complètement intégrée dans la chaîne CAO d'une manière quasi transparente à l'utilisateur.

Nous pensons que la simulation numérique, à terme, sera absorbée par des outils de conception mêlant l'intelligence artificielle au calcul scientifique. Nous espérons que notre travail aura contribué à cette dynamique en tant qu'étape vers un système de conception de modélisations complexes mélangeant des phénomènes physiques et des méthodes numériques de différentes natures.

- [1] C. ALESSANDRI, A. TRALLI  
"An Alternative Technique for Reducing Domain Integrals to the Boundary" in  
Boundary Element in Mechanical and Electrical Engineering. Proceedings of the  
International BE Symposium, Nice, France 15-17 May 1990. Comp. Mech. Publ.
- [2] P.K. BANERJEE, R. BUTTERFIELD  
"Boundary Element Method in Engineering Science"  
McGraw-Hill, London 1981
- [3] H. BLUM  
"Numerical Treatment of corner and crack singularities" in  
Finite element and boundary element techniques from mathematical and engineering  
point of view". CISM Courses and Lectures n°301.  
Springer Verlag 1988. pp 171-212
- [4] C.A. BREBBIA, J.C.F. TELLES, L.C. WROBEL  
"Boundary Element Techniques"  
Springer-Verlag, Berlin Heidelberg New York Tokyo 1984
- [5] C.A. BREBBIA  
"Basis of boundary elements"  
in Progress in boundary element method  
Pentech Press Ltd 1981
- [6] C.A. BREBBIA  
"The boundary element method for engineers"  
Pentech Press, London 1978
- [7] C.A. BREBBIA, J.C. CONNOR  
"Boundary integral formulation"  
in Topics in Boundary element research (Vol 1)  
Springer-Verlag, Berlin Heidelberg New York Tokyo 1984
- [8] C.A. BREBBIA, L. WROBEL  
"Non-Linear Transient Thermal Analysis Using the Dual Reciprocity Method" in  
Boundary Element Techniques : Applications in Stress Analysis and Heat Transfer  
Computational Mechanics Publication 1987
- [9] E. BRUCH, A. LEJEUNE, B. MERGEAI  
"Application of the BEM to Free Surface Seepage  
from Canals to Anisotropic Soils" in  
Boundary Element in Mechanical and Electrical Engineering. Proceedings of the  
International BE Symposium, Nice, France 15-17 May 1990. Comp. Mech. Publ.
- [10] E. BRUCH  
"Contribution à la résolution de l'équation de Laplace par la méthode des éléments  
frontières avec application aux écoulements dans les milieux poreux"  
Thèse en Sciences Appliquées, Université de Liège (Belgique), 1989
- [11] M. BURNET-FAUCHEZ  
"The use of boundary element finite element  
coupling method in electrical engineering" in  
Topics in Boundary Element Research. Vol 6 (Electromagnetic Applications)  
Springer Verlag 1989

- [12] E. CANOT  
"Généralisation de la méthode intégrale aux frontières pour les systèmes fluide-fluide"  
Thèse INPG Grenoble 1989. Spécialité Mécanique
- [13] M. CHAUDONNERET  
"Sur les discontinuités du vecteur contrainte dans les calculs de structure par la méthode des équations intégrales" in  
C.R. Acad. Science Paris, t. 284 (21 février 1977), Série A pp 463-466
- [14] T.A. CRUSE, F.J. RIZZO  
"A direct formulation and numerical simulation of the general Transcient elasto-dynamic problem"  
J. Math. Analysis Applic. n°22 1968
- [15] T.A. CRUSE, R. AITHAL  
"Non-Singular boundary integral equation implementation"  
Int. Jour. Num. Meth. Eng., Vol 36 (1993), pp 237-254
- [16] G.DHAT, G. TOUZOT  
"Une présentation de la méthode des éléments finis"  
MALOINE 1984
- [17] DT2i  
Manuel d'utilisation de Flux-Expert  
DT2i, 38240 MEYLAN ZIRST
- [18] W.R. DYKSEN, C.R. GRITTER  
"Elliptic Expert : an expert system for elliptic partial differential equations"  
Mathematics and computer in simulation, Vol 31, 1989, pp 333-342
- [19] B. FENGSHENG, L. JIAQI  
"A new numerical algorithm for solving fundamentals solutions" in  
Boundary Element IX, Vol 1. Springer Verlag 1987
- [20] B.A. FINLAYSON  
"The method of weighted residuals and variational principles"  
Academic Press N.Y. 1972
- [21] B. FORD, S.J. HAGUE, R.M.J. ILES  
"Numerical knowledge-based systems"  
Mathematics and computer in simulation, Vol 31, 1989, pp 395-400
- [22] L.J. GRAY, E. LUTZ  
"On the treatment of corners in the boundary element method" in  
Journal of Computational and Applied Mechanics, Vol 32 (1990). pp 369-386
- [23] P. GRISVARD  
"Problèmes aux limites dans les polygones. Mode d'emploi"  
EDF. Bulletin de la DER Série C. N°1. 1986. pp 21-59
- [24] M. GUIGGIANI, A. GIGANTE  
"A General Algorithm for multidimensional Cauchy principal value integrals in the boundary element method"  
Journal of Applied Mechanics, Vol 57 (1990), pp 906-915
- [25] K. HAYAMI, C.A. BREBBIA  
"A new coordinate transformation for singular and nearly singular integrals over general curved boundary elements" in  
Boundary element IX, Vol 1. Springer Verlag 1987



- [26] D.P. HENRY, P.K. BANERJEE  
"A new Boundary Element Formulation for two and three dimensional Thermoelasticity using particular integrals"  
Int. Jour. Num. Meth. Eng., Vol 26 (1988) pp 2061-2077
- [27] HONG-BAO LI, GUO-MING HAN, HERBERT A. MANG  
"A New method for evaluating singular integrals in stress analysis of solids by the direct boundary element method"  
Int. Jour. Num. Meth. Eng., Vol 21 (1985), pp 2071-2098
- [28] T.V. HROMADKA II  
"Best approximation of a linear operation equation" in  
Boundary Element IX, Vol 1. Springer Verlag 1987
- [29] J.F. IMHOFF  
"Modélisation magnétique et mécanique des machines électriques par la méthode des éléments finis"  
Thèse INPG Grenoble 1989. Génie électrique
- [30] D.B. INGHAM, M.A. KELMANSON  
"Solution of nonlinear elliptic equations with boundary singularities by an integral equation method" in  
Boundary Element V. Computational Mechanics, Springer Verlag 1985
- [31] M.A. JASWON  
"Integral equation methods in potential theory. I"  
Proc. Roy. Soc. London. Ser. A. Math. Phys. Sc. Vol 275 (1963) pp 23-32
- [32] Y. JIANGSHENG, X. Xi, S. HANGANG  
"Unified boundary integral equation and unified boundary element equation"  
IEEE Transactions on magnetics, Vol 26, n°2 Mars 1990
- [33] Y. KATAYAMA, M. KODA  
"Scaling integration method for singular integrals and its application to BEM"  
Journal of computational physics, Vol 83 (1989), pp 292-302
- [34] M.A. KEAVEY  
"Declarative language techniques for computational software"  
Numerical Methods in Thermal Problems.  
Vol VI, Part 2, Pineridge Press 1989, pp 1131-1141
- [35] M.A. KELMANSON  
"Solution of nonlinear elliptic equation with boundary singularities by an integral equation method" in  
Journal of Computational Physics, Vol 56 (1984). pp 244-258
- [36] A. KOST, J.X. SHEN  
"Treatment of singularities in the computation of magnetics fields with periodic boundary conditions by the boundary element method"  
IEEE Trans. on Magnetics, Vol 26, n°2, Mars 1990
- [37] L. KRÄHENBÜHL  
"La méthode des équations intégrales de frontières pour la résolution des problèmes de potentiel en electrotechnique, et sa formulation en axisymétrique"  
Thèse ECL. Departement Electrotechnique Lyon 1983
- [38] D.A. LOWTHER, C.M. SALDANHA, G. CHOY  
"The applications of Expert-System to CAD in electromagnetics"  
IEEE Trans. on Magnetics, Vol MAG 21, n°6, Nov. 1985

- [39] P. MASSÉ  
"Analyse méthodologique de la modélisation numérique des équations de la physique des milieux continus à l'aide de la méthode des éléments finis. Flux-Expert : un système d'aide à la construction de logiciels."  
Thèse d'état INPG. Grenoble 1983
- [40] D.E. MEDINA, M.H. LEAN, J.A. LIGETT  
"Boundary Element Method in zoned media : direct and indirect methods"  
Int. Jour. Num. Meth. Eng., Vol 29 (1990), pp 1727-1735
- [41] A.K. MITRA, M.S. INGSBER  
"Resolving Difficulties in the BIEM caused by geometric corners and discontinuous boundary conditions" in  
Boundary Element IX, Vol 1, Springer Verlag, 1987
- [42] D. NARDINI, C.A. BREBBIA  
"A new approach to free vibration analysis using boundary elements" in  
Boundary Element Method in Engineering. C.A. Brebbia (Ed.)  
Springer Verlag (1982)
- [43] P.W. PARTRIDGE, C.A. BREBBIA  
"Computer implementation of the BEM Dual reciprocity method for the solution of general field equations"  
Communications in Applied Numericals Methods, Vol 6 (1990) pp 83-92
- [44] R.L. PESKIN, S.S. WALTHER, A.M. FRONCIONI  
"Smalltalk, the next generation scientific computing interface"  
Mathematics and computer in simulation, Vol 31, 1989, pp 371-381
- [45] C. PATTERSON, M.A. SHEIKH  
"Discontinuous Boundary Element for heat conduction" in  
Numerical Methods in thermal problems, Vol II. Pineridge Press 1981
- [46] C PATTERSON, N.A.S. EL SEBAI  
"A Family of partially discontinuous boundary elements for three dimensional analyses" in  
Boundary Element V. Comp. Mech., Springer Verlag 1983. pp 193-206
- [47] F.J. RIZZO, D.J. SHIPPY  
"An Advanced Boundary integral Equation Method for three-dimensional Thermoelasticity"  
Int. Jour. Num. Meth. Eng, Vol 11 (1977) pp 1753-1768
- [48] G.F. ROACH  
"Green's functions"  
Second edition, Cambridge University Press 1982
- [49] J.W.ROZENBLIT, J. WU  
"Integrated knowledge representation and management in simulation-based design generation"  
Mathematics and computer in simulation, Vol 34, 1992, pp 361-382
- [50] I. STACKGOLD  
"Green's functions and boundary value problems"  
Wiley Interscience 1979
- [51] G.T. SYMM  
"Integral equation methods in potential theory. II"  
Proc. Roy. Soc. London. Ser. A. Math. Phys. Sc. Vol 275 (1963) pp 33-46

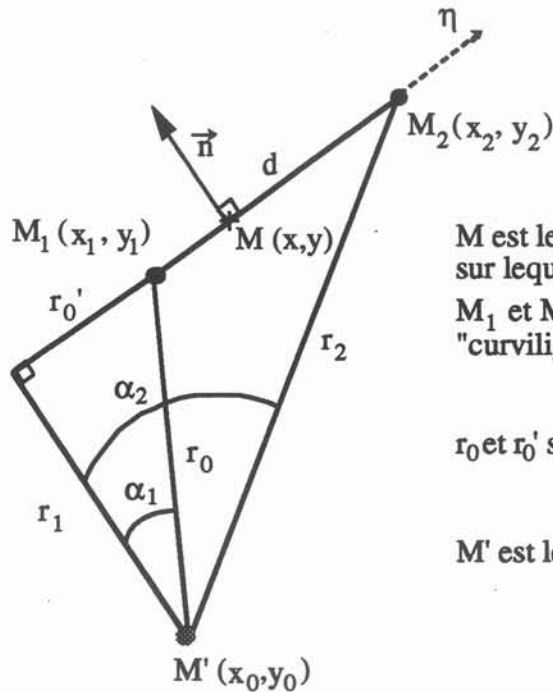
- [52] W. TANG, C.A. BREBBIA, J.C.F. TELLES  
 "A Generalised Approach to Transfer the Domain Integrals onto Boundary Ones for Potential Problems in BEM" in  
 Boundary Element IX, Vol 1. Computational Mechanics Publications.  
 Springer Verlag 1987
- [53] C.P. TONG, T OMEDA, Y. KIMURA  
 "Application of Boundary Element Method in two and three dimensional unsteady heat transfer problems involving phase change, solidification problems." in  
 Boundary Element V. Comp.Mech., Springer Verlag 1983, pp 153-162
- [54] E. TONTI  
 "The reason for analogies between physical theories"  
 Appl. Math. Modelling, vol 1 June 1976, pp 37-50
- [55] N TOSAKA, K. KAKUDA  
 "The Generalized Boundary Element Method for Non linear Problems" in  
 Boundary Element X, Vol 1. Springer Verlag 1988
- [56] S. WALKER  
 "Fundamental solutions"  
 in Progress in boundary element method  
 Pentech Press Ltd 1981
- [57] W.L. WENDLAND  
 "On asymptotic error estimates for combined BEM and FEM" in  
 Finite Element and Boundary Element techniques from mathematical and engineering point of view. CISM Courses and Lectures n° 301.  
 Springer-Verlag. 1988. pp 273-333
- [58] J.C. WU  
 "Boundary Element Solution of Viscous Flow Problems" in  
 Boundary Element Techniques : Application in Fluid Flow and Computational Aspects. Computational Mechanics Publication 1987
- [59] WU ZHANG, HUAN-RAN XU  
 "A General and effective way for evaluating the integrals with various orders of singularity in the direct boundary element method"  
 Int. Jour. Num. Meth. Eng., Vol 28 (1989), pp 2059-2064
- [60] ZHENG JIA DONG  
 "Location of free surface for zoned problems  
 A new economical boundary element method" in  
 Boundary Element V. Comp.Mech., Springer Verlag 1983, pp 85-94
- [61] O.C. ZIENKIEWICZ, D.W. KELLY, P BETTES  
 "The coupling of the finite element method and boundary solution procedures"  
 Int. Jour. Num. Meth. Eng., Vol 11 (1977), pp 355-375
- [62] O.C. ZIENKIEWICZ, R.L. TAYLOR  
 "La méthode des éléments finis - Formulation de base et problèmes linéaires"  
 (Traduction Française) AFNOR Technique 1991

Il s'agit de calculer les intégrales intervenant dans la formulation MIF de l'équation de Laplace 2D avec des éléments numériques à l'ordre zéro, c'est à dire sur lesquels la variable est constante, on va donc calculer analytiquement :

$$H = \int_{\text{segment}} \frac{\partial \text{Ln}(M'M)}{\partial n} (dM)$$

$$G = \int_{\text{segment}} \text{Ln}(M'M) (dM)$$

On peut faire le schéma suivant :



M est le point courant sur le segment sur lequel on intègre entre les points \$M\_1\$ et \$M\_2\$, \$\eta\$ est l'abscisse "curviligne" le long du segment

\$r\_0\$ et \$r\_0'\$ sont signés

\$M'\$ est le pôle de la fonction de Green

On a alors deux cas bien distincts :

- Intégrales régulières (C'est le cas ou \$M' \notin\$ segment)

#### Calcul de H

On a :

$$\frac{\partial \text{Ln}(M'M)}{\partial n} = \text{Grad} (\text{Ln}(M'M)) \cdot \vec{n}$$

or 
$$\text{Grad} (\text{Ln}(\overrightarrow{M'M})) = \frac{\overrightarrow{M'M}}{(\overrightarrow{M'M})^2}$$

Si l'on se réfère au schéma effectué, on a noté :

$$\overrightarrow{M'M} \cdot \vec{n} = r_0$$

On peut donc écrire

$$H = \int_{r_0'}^{r_0'+d} \frac{r_0}{r_0^2 + \eta^2} d\eta = \left[ \text{Arctg}\left(\frac{\eta}{r_0}\right) \right]_{r_0'}^{r_0'+d}$$

Soit : 
$$H = \alpha_2 - \alpha_1$$

### Calcul de G

Avec les mêmes notations que précédemment, on peut écrire :

$$G = \int_{r_0'}^{r_0'+d} \text{Ln}(\sqrt{r_0^2 + \eta^2}) d\eta$$

$$G = \frac{1}{2} \left[ \eta \text{Ln}(r_0^2 + \eta^2) - 2\eta + 2r_0 \text{Arctg}\left(\frac{\eta}{r_0}\right) \right]_{r_0'}^{r_0'+d}$$

Soit : 
$$G = (r_0' + d) \text{Ln}(r_2) - (r_0' + d) + r_0 \alpha_2 - (r_0' \text{Ln}(r_1) - r_0' + r_0 \alpha_1)$$

### • Intégrales singulières (C'est le cas où $M' \in \text{segment}$ )

Les deux expressions ci-dessus dégénèrent très bien, on peut donc en déduire les valeurs des intégrales singulières. On rappelle que l'on est dans le cas particulier où  $M'$  est au milieu du segment.

### Calcul de H

Dans ce cas  $r_0 = 0$  et donc :

$$H = 0$$

### Calcul de G

Dans ce cas  $r_0' = -\frac{d}{2}$  et  $r_1 = r_2 = \frac{d}{2}$  et donc

$$G = d \left( \text{Ln}\left(\frac{d}{2}\right) - 1 \right)$$

Il s'agit de calculer les intégrales intervenant dans la formulation MIF de l'équation de Laplace 2D avec des éléments numériques à l'ordre 1, c'est à dire sur lesquels la variable est linéaire, on va donc calculer analytiquement :

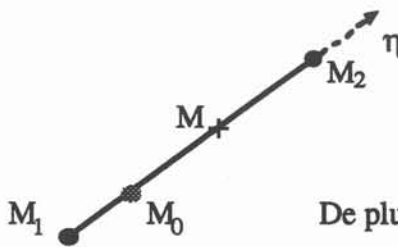
$$H = \int_{\text{segment}} \frac{\partial \text{Ln}(M_0 M)}{\partial n} T (dM)$$

$$G = \int_{\text{segment}} \text{Ln}(M_0 M) \frac{\partial T}{\partial n} (dM)$$

où  $M_0$  appartient au segment

$T$  et  $\partial T / \partial n$  sont des inconnues qui varient linéairement sur le segment, on notera par la suite  $X$ , l'une quelconque de ces inconnues.

On peut faire le schéma suivant :



$M$  est le point courant sur le segment sur lequel on intègre entre les points  $M_1$  et  $M_2$

$M_0$  est le pôle de la fonction de Green

De plus, on note :  $D_1$ , la distance  $\overline{M_0 M_1}$

$D_2$ , la distance  $\overline{M_0 M_2}$

### Calcul de H

On a, exactement comme pour le calcul de l'annexe A1 :

$$\frac{\partial \text{Ln}(M_0 M)}{\partial n} = \text{Grad} (\text{Ln}(M_0 M)) \cdot \vec{n} = \frac{\overrightarrow{M_0 M}}{(M_0 M)^2} \cdot \vec{n}$$

Or, il est clair que, dans ce cas où  $M_0 \in \text{segment}$ , on a :

$$\overrightarrow{M_0 M} \cdot \vec{n} = 0$$

On peut donc écrire :

$$H = 0$$

### Calcul de G

On cherche donc à intégrer :



$$G = \int_{M_1}^{M_2} \text{Ln}(\sqrt{(x-x_0)^2 + (y-y_0)^2}) X(x,y) dl \quad (dl = \sqrt{dx^2 + dy^2})$$

Nous allons séparer cette intégrale en deux intégrales, l'une entre  $M_0$  et  $M_1$ , et l'autre entre  $M_0$  et  $M_2$ . Ces deux intégrales, si l'on fait un changement de variable ramenant l'intervalle entre 0 et 1, sont formellement identiques. En effet, si  $X$  varie linéairement entre  $M_1$  et  $M_2$ , la variation est aussi linéaire sur chacun des deux intervalles. Calculons par exemple l'intégrale entre  $M_0$  et  $M_1$ , c'est à dire :

$$G_1 = \int_0^1 \text{Ln}(D_1 \eta) X(\eta) D_1 d\eta$$

avec  $X(\eta) = \eta X_1 + (1-\eta) X_0$ , or grâce à l'hypothèse de variation linéaire, on peut écrire :

$$X_0 = \frac{D_1 X_2 + D_2 X_1}{D_1 + D_2}$$

En effectuant un deuxième changement de variable,  $z=D_1 \eta$ , on peut écrire :

$$G_1 = \frac{X_1}{D_1+D_2} \int_0^{D_1} (D_2+z) \text{Ln}(z) dz + \frac{X_2}{D_1+D_2} \int_0^{D_1} (D_1-z) \text{Ln}(z) dz$$

Les deux expressions ci-dessus sont parfaitement intégrables analytiquement et l'on obtient :

$$\begin{aligned} G_1 = & X_1 \cdot \frac{D_1}{D_1+D_2} (D_2 \text{Ln}(D_1) - D_2 + \frac{D_1}{2} \text{Ln}(D_1) - \frac{D_1}{4}) \\ & + X_2 \cdot \frac{D_1}{D_1+D_2} (D_1 \text{Ln}(D_1) - D_1 - \frac{D_1}{2} \text{Ln}(D_1) + \frac{D_1}{4}) \end{aligned}$$

Il suffit d'inverser les indices 1 et 2 partout pour obtenir l'expression de  $G_2$ . Finalement, on a donc :

$$\begin{aligned} G = & X_1 \cdot \frac{1}{D_1+D_2} \left\{ D_1 D_2 (\text{Ln}(D_1)-1) + \frac{D_2^2}{2} \text{Ln}(D_2) + \frac{D_1^2}{2} (\text{Ln}(D_1) - \frac{1}{2}) - \frac{3}{4} D_2^2 \right\} \\ & + X_2 \cdot \frac{1}{D_1+D_2} \left\{ D_1 D_2 (\text{Ln}(D_2)-1) + \frac{D_1^2}{2} \text{Ln}(D_1) + \frac{D_2^2}{2} (\text{Ln}(D_2) - \frac{1}{2}) - \frac{3}{4} D_1^2 \right\} \end{aligned}$$

Remarque :

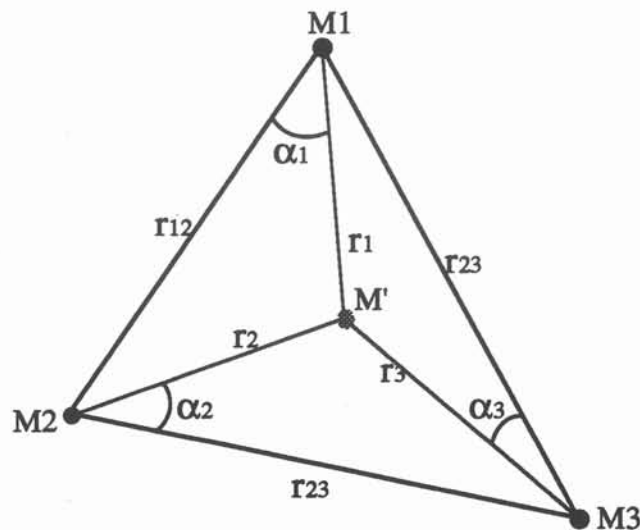
L'expression ci-dessus dégénère très bien si  $D_1$  ou  $D_2$  est nulle, c'est à dire si le pôle est confondu avec l'un des deux nœuds, on a :

$$\text{Si } D_1=0, \quad G = X_1 \cdot \frac{D_2}{2} (\text{Ln}(D_2) - \frac{3}{2}) + X_2 \cdot \frac{D_2}{2} (\text{Ln}(D_2) - \frac{1}{2})$$

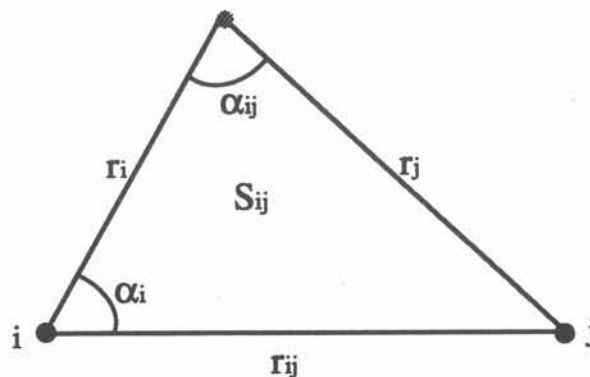
$$\text{Si } D_2=0, \quad G = X_1 \cdot \frac{D_1}{2} (\text{Ln}(D_1) - \frac{1}{2}) + X_2 \cdot \frac{D_1}{2} (\text{Ln}(D_1) - \frac{3}{2})$$

Il s'agit d'intégrer l'intégrale singulière intervenant dans la formulation MIF de l'équation de Laplace 3D avec des éléments numériques à l'ordre zéro, c'est à dire sur lesquels la variable est constante, on veut donc intégrer  $G_{M'}(M) = \frac{1}{M'M}$  sur un triangle, sachant que  $M'$  appartient

au triangle. On fait le schéma suivant :



et l'on va intégrer sur chacun des sous-triangles ainsi formés. Il y a donc trois intégrales de  $\frac{1}{r}$  similaires à calculer, chacune sur des triangles dont l'origine est un des sommets, soit le schéma suivant :



Or, l'intégrale de  $\frac{1}{r}$  dans cette configuration est connue, on la trouve notamment dans [BREBBIA] (on la retrouve facilement en passant en coordonnées polaires), c'est :

$$I(i,j) = \frac{2S_{ij}}{r_{ij}} \operatorname{Ln} \left( \frac{\operatorname{tg} \left( \frac{\alpha_i + \alpha_{ij}}{2} \right)}{\operatorname{tg} \left( \frac{\alpha_i}{2} \right)} \right)$$

- $2S_{ij}$  se calcule simplement puisque l'on a l'égalité :

$$2 S_{ij} = \| \vec{r}_i \wedge \vec{r}_j \|$$

- Quant aux angles, on peut utiliser la formules des cosinus, c'est à dire par exemple pour  $\alpha_{ij}$  :

$$\cos ( \alpha_{ij} ) = \frac{r_i^2 + r_j^2 - r_{ij}^2}{2 r_i r_j}$$

Pour calculer l'intégrale qui nous intéresse, il suffit donc d'ajouter les trois intégrales  $I(i,j)$  issues des trois sous-triangles. On pourra vérifier que les formules sont aussi valables si l'origine est en bordure du triangle, à condition de remarquer la nullité d'au moins une (deux si l'origine est confondue avec un sommet du triangle) des intégrales  $I(i,j)$ .

# Thèse de Doctorat de l'Institut National Polytechnique de Grenoble

Auteur : Patrick SZTULZAF

Établissement : Institut National Polytechnique de Grenoble

## Titre

GREEN-EXPERT  
UN SOLVEUR GÉNÉRALISÉ ASSOCIÉ À  
UN GÉNÉRATEUR DE FORMULATIONS  
POUR LA MÉTHODE DES INTÉGRALES  
DE FRONTIÈRES.

## Title

GREEN-EXPERT  
A GENERAL SOLVER COMBINED WITH A  
FORMULATIONS GENERATOR FOR THE  
BOUNDARY ELEMENT METHOD.

## Résumé

De nombreux secteurs de l'industrie et de la recherche utilisent la modélisation des phénomènes de la physique des milieux continus. Les équations aux dérivées partielles décrivant ces phénomènes sont résolues à l'aide de diverses méthodes numériques. Les modélisations utilisées sont de plus en plus pointues, tant au niveau physique qu'au niveau numérique. Les réponses logicielles à ces problèmes doivent donc être évolutives.

Ce travail s'insère dans une dynamique de recherche dans le domaine de la modélisation des phénomènes complexes qui a débuté avec l'élaboration du programme Flux-Expert®, basé sur la Méthode des Éléments Finis. Afin d'élargir le champ des possibilités offertes par ce programme, nous avons choisi d'y associer la Méthode des Intégrales de Frontières.

Dans cette optique, après une présentation didactique de la Méthode des Intégrales de Frontières, nous proposons une décomposition générale de la résolution numérique d'un problème à l'aide de cette méthode.

Nous décrivons ensuite le logiciel issu de cette analyse : Green-Expert. L'originalité de la démarche réside dans l'association d'un programme Générateur de Formulations et d'un programme Solveur généralisé. Ce Solveur est capable de résoudre tout problème décrit à l'aide d'une formulation issue du Générateur et d'une Géométrie discrétisée.

La dernière partie de ce mémoire est consacrée à la validation. Des exemples de couplage entre la Méthode des Intégrales de Frontières et la Méthode des Éléments Finis sont présentés. Enfin, des exemples de résolution 2D et 3D permettent de valider le Générateur et le Solveur de Green-Expert.

## Abstract

Investigations in many sectors of industry and research require the modelling of phenomena observed in the physics of continuous media. The partial differential equations describing these phenomena are solved using a wide range of numerical methods. The models used are increasingly sophisticated, from both a physical and numerical point of view. Software used to solve these problems must therefore be capable of evolving.

This work is a continuation of research efforts devoted to the modelling of complex phenomena that began with the development of the Flux-Expert® program, based on the Finite Element Method. In order to extend the possibilities offered by this program, we decided to combine it with the Boundary Element Method.

After reviewing the Boundary Element Method, we propose a general decomposition of the numerical solution of a problem using this method.

We then describe the Green-Expert software developed on the basis of this analysis. The original aspect of the approach lies in the combination of a formulations generator and a general solver. This solver is capable of solving any problem described using a formulation coming from the Generator and a discrete geometry.

The last part of this thesis is devoted to the validation phase. Examples of the combined use of the Boundary Element and the Finite Element Methods are presented and examples of 2D and 3D resolution are used to validate the Green-Expert Solver and Generator.

## Mots clés

Analyse méthodologique; Modélisation numérique; Méthodes numériques; Couplage; Méthode des intégrales de frontières; Méthode des éléments finis; Générateur de formulations; Fonctions de Green



## AUTORISATION DE SOUTENANCE

Vu les dispositions de l'arrêté du 30 Mars 1992 relatifs aux Etudes Doctorales

Vu les Rapports de présentations de :

Monsieur Adel RAZEK (Directeur de Recherche)

Monsieur Laurent KRAHENBUHL (Chargé de Recherche)  
(co-signataire : Monsieur Alain NICOLAS, Professeur)

**Monsieur Patrick SZTULZAIT**

est autorisé à présenter une thèse en soutenance en vue de l'obtention du diplôme de  
Docteur de l'Institut National Polytechnique de Grenoble, spécialité  
"Energétique Physique".

Fait à Grenoble, le 24 Janvier 1994

Pour le Directeur de l'INPG  
le Directeur des Etudes Doctorales



Thèse de Doctorat de l'Institut National  
Polytechnique de Grenoble

Auteur Patrick SZTULZAF  
Établissement Institut National  
Polytechnique de Grenoble

*Titre*

GREEN-EXPERT  
UN SOLVEUR GÉNÉRALISÉ ASSOCIÉ À UN  
GÉNÉRATEUR DE FORMULATIONS POUR LA  
MÉTHODE DES INTÉGRALES DE  
FRONTIÈRES.

*Résumé*

De nombreux secteurs de l'industrie et de la recherche utilisent la modélisation des phénomènes de la physique des milieux continus. Les équations aux dérivées partielles décrivant ces phénomènes sont résolues à l'aide de diverses méthodes numériques. Les modélisations utilisées sont de plus en plus pointues, tant au niveau physique qu'au niveau numérique. Les réponses logicielles à ces problèmes doivent donc être évolutives.

Ce travail s'insère dans une dynamique de recherche dans le domaine de la modélisation des phénomènes complexes qui a débuté avec l'élaboration du programme Flux-Expert®, basé sur la Méthode des Éléments Finis. Afin d'élargir le champ des possibilités offertes par ce programme, nous avons choisi d'y associer la Méthode des Intégrales de Frontières.

Dans cette optique, après une présentation didactique de la Méthode des Intégrales de Frontières, nous proposons une décomposition générale de la résolution numérique d'un problème à l'aide de cette méthode.

Nous décrivons ensuite le logiciel issu de cette analyse : Green-Expert. L'originalité de la démarche réside dans l'association d'un programme Générateur de Formulations et d'un programme Solveur généralisé. Ce Solveur est capable de résoudre tout problème décrit à l'aide d'une formulation issue du Générateur et d'une Géométrie discrétisée.

La dernière partie de ce mémoire est consacrée à la validation. Des exemples de couplage entre la Méthode des Intégrales de Frontières et la Méthode des Éléments Finis sont présentés. Enfin, des exemples de résolution 2D et 3D permettent de valider le Générateur et le Solveur de Green-Expert.

*Mots clés*

Analyse méthodologique; Modélisation numérique; Méthodes numériques; Couplage; Méthode des intégrales de frontières; Méthode des éléments finis; Générateur de formulations; Fonctions de Green.